



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

**Desarrollo de un sistema de administración de cuentas
para mejorar el control de acceso orientado al sistema
Quipuc de una empresa de consultoría de software**

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniero de Software

AUTOR

Juan Inocente GUERRERO RODRÍGUEZ

ASESOR

Frank Edmundo ESCOBEDO BAILÓN

Lima, Perú

2018



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Guerrero, J. (2018). *Desarrollo de un sistema de administración de cuentas para mejorar el control de acceso orientado al sistema Quipuc de una empresa de consultoría de software*. Trabajo de Suficiencia Profesional para optar el título de Ingeniero de Software. Escuela Profesional de Ingeniería de Software, Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima, Perú.

Hoja de metadatos complementarios

Código ORCID del autor	—
DNI o pasaporte del autor	45001522
Código ORCID del asesor	0000-0002-2058-0976
DNI o pasaporte del asesor	41671087
Grupo de investigación	—
Agencia financiadora	Autofinanciado
Ubicación geográfica donde se desarrolló la investigación	<p>Lugar:</p> <p>País: Perú Departamento: Lima Provincia: Lima Distrito: Pueblo Libre Calle: Alfredo Maldonado Nro. 654</p> <p>Coordenadas geográficas: -12.074415, -77.060578</p>
Año o rango de años en que se realizó la investigación	2018
Disciplinas OCDE	Ingeniería de sistemas y comunicaciones http://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
Escuela Profesional de Ingeniería de Sistemas

**Acta de Sustentación del
Trabajo de Suficiencia Profesional**

Siendo las ~~20~~ 22 horas del día ~~20~~ de diciembre del año 2018, se reunieron los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Ing. Carrasco Ore Nilo Eloy (Presidente), Ing. Contreras Flores Walter Pedro (Miembro) y el Dr. Escobedo Bailón Frank Edmundo (Miembro Asesor) para la sustentación del Trabajo de Suficiencia Profesional Intitulado: **"DESARROLLO DE UN SISTEMA DE ADMINISTRACIÓN DE CUENTAS PARA MEJORAR EL CONTROL DE ACCESO ORIENTADO AL SISTEMA QUIPUC DE UNA EMPRESA DE CONSULTORÍA DE SOFTWARE"**, por el Bachiller: **Guerrero Rodríguez, Juan Inocente**; para obtener el Título Profesional de Ingeniero de Software.

Acto seguido de la exposición del Trabajo de Suficiencia Profesional, el Presidente invitó al Bachiller a dar las respuestas a las preguntas establecida por los miembros del Jurado.

El Bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, el Bachiller obtuvo la nota de ~~15~~ (En letras).... **QUINCE**.....

A continuación el presidente del jurado el Ing. Carrasco Ore Nilo Eloy, declara al Bachiller Ingeniero de Software.

Siendo las ~~21~~ 22 horas, se levantó la sesión.

Presidente

Ing. Carrasco Ore Nilo Eloy

Miembro

Ing. Contreras Flores Walter Pedro

Miembro Asesor

Dr. Escobedo Bailón Frank Edmundo

FICHA CATALOGRÁFICA

**DESARROLLO DE UN SISTEMA DE ADMINISTRACIÓN DE CUENTAS
PARA MEJORAR EL CONTROL DE ACCESOS AL SISTEMA QUIPUC DE
UNA EMPRESA DE CONSULTORÍA DE SOFTWARE**

AUTOR: GUERRERO RODRÍGUEZ, JUAN INOCENTE

ASESOR: ESCOBEDO BAILÓN, FRANK EDMUNDO

LIMA – PERÚ, 2018

Título profesional: Ingeniero de Software

Área / Programa / Línea de Investigación:

Ingenierías / Tecnología de Información y Comunicación / Ingeniería de Software

**Pregrado: Universidad Nacional Mayor de San Marcos – Facultad de Ingeniería de
Sistemas e Informática – Escuela Profesional de Ingeniería de Software.**

Formato 28 x 20 cm

Páginas: xi, 107

Este trabajo se lo dedico a mi querida madre Alicia,
por cuidarme y protegerme, a mis hermanos,
por apoyarme y aconsejarme, y a mi padre Luis, abuela
y abuelo que me cuidan desde el cielo.

AGRADECIMIENTOS

Quiero agradecer a mi madre y hermanos, por el apoyo continuo e incondicional que me han brindado a lo largo de mi vida.

A mi familia, por cuidarme y aconsejarme en cada paso que daba en la vida.

A mis amigos, por brindarme su confianza y ánimos en cada etapa de mi vida académica.

A lo profesores de la universidad, por todo el conocimiento brindado durante todos los años que estuve en la universidad.

A mi asesor, el profesor Frank Escobedo, por su orientación y consejos para elaborar exitosamente este trabajo.

A todas aquellas personas que, a pesar de no conocerme, me dieron consejos que muchas veces infravalore en su momento pero con el tiempo fui tomando en serio.

**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SOFTWARE**

**DESARROLLO DE UN SISTEMA DE ADMINISTRACIÓN DE CUENTAS
PARA MEJORAR EL CONTROL DE ACCESOS AL SISTEMA QUIPUC DE
UNA EMPRESA DE CONSULTORÍA DE SOFTWARE**

Autor: Guerrero Rodríguez, Juan Inocente
Asesor: Escobedo Bailón, Frank Edmundo
Título: Informe de Trabajo de Suficiencia Profesional para optar el Título Profesional de Ingeniero de Software
Fecha: Diciembre 2018

RESUMEN

El presente informe describe uno de los proyectos que más ha aportado en el crecimiento profesional del autor. Con ayuda de la Metodología Ágil Scrum se logró realizar un trabajo acorde a las necesidades del Product Owner, mejorando más la confianza entre el equipo de trabajo y la empresa. El problema que aborda en el presente informe trata sobre el control de acceso de las cuentas del sistema Quipuc. Se requiere que el control se realice mediante un sistema que maneje los estados asignados a la cuenta, debiendo ser actualizados mediante procesos automáticos y monitoreados por el administrador. En el desarrollo de la solución se detallan los pasos a seguir para la implementación del Sistema de Administración de Cuentas, siguiendo la metodología SCRUM. Esto permitió reducir tiempos de desarrollo, realizar cambios de manera más rápida y realizar presentaciones periódicas del producto al Product Owner. Uno de los logros obtenidos fue la de afianzar la confianza entre el equipo de trabajo con la empresa, ya que se cumplió con las expectativas planteadas. Además, se logró generar un ingreso económico para la empresa.

Palabras claves: administración cuentas, aplicación web, procesos automáticos, meanjs, SCRUM.

**MAJOR NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS ENGINEERING
PROFESSIONAL SCHOOL OF SOFTWARE ENGINEERING**

**DEVELOPMENT OF AN ACCOUNTS MANAGEMENT SYSTEM TO
IMPROVE THE CONTROL OF ACCESS TO THE QUIPUC SYSTEM OF A
SOFTWARE CONSULTING COMPANY**

Author: Guerrero Rodríguez, Juan Inocente
Advisor: Escobedo Bailón, Frank Edmundo
Title: Professional Sufficiency Work Report for opt for the Professional Title of
Software Engineer
Date: December 2018

ABSTRACT

This report describes one of the projects that has contributed most to the author's professional growth. With the help of the Agile Methodology Scrum was achieved perform work according to the needs of the Product Owner, improving the trust between the work team and the company. The problem addressed in this report is about control of access of Quipuc system accounts. It's required that the control is done through a system that manages the states assigned to the account, and must be updated through automatic processes and monitored by the administrator. In the development of the solution, the steps to be followed for the implementation of the Account Management System are detailed, following the SCRUM methodology. This allowed to reduce development times, make changes more quickly and make periodic presentations of the product to the Product Owner. One of the achievements was to strengthen the trust between the work team and the company, since the expectations were met. In addition, it was possible to generate an economic income for the company.

Key words: accounts management, web application, automatic processes, means, SCRUM.

TABLA DE CONTENIDO

CARÁTULA	i
FICHA CATALOGRÁFICA	ii
DEDICATORIA	iii
AGRADECIMIENTOS	iv
RESUMEN	v
ABSTRACT	vi
ÍNDICE O TABLA DE CONTENIDOS	vii
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS	xi
INTRODUCCIÓN	1
CAPÍTULO I – TRAYECTORIA PROFESIONAL	3
1.1 PRESENTACIÓN PROFESIONAL	3
1.2 EXPERIENCIA PROFESIONAL	3
1.3 FORMACIÓN PROFESIONAL	5
1.4 IDIOMAS	5
1.5 OTROS CONOCIMIENTOS	5
CAPÍTULO II – CONTEXTO EN EL QUE SE DESARROLLÓ LA EXPERIENCIA	6
2.1 EMPRESA - ACTIVIDAD QUE REALIZA	6
2.2 VISIÓN	9
2.3 MISIÓN	9
2.4 ORGANIZACIÓN DE LA EMPRESA	9
2.5 ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS	10
2.6 EXPERIENCIA PROFESIONAL REALIZADA EN LA ORGANIZACIÓN	10
CAPÍTULO III – ACTIVIDADES DESARROLLADAS	14
3.1 SITUACIÓN PROBLEMÁTICA	14
3.1.1 DEFINICIÓN DEL PROBLEMA	14
3.2 SOLUCIÓN	15
3.2.1 OBJETIVOS	16
3.2.2 ALCANCE	17
3.2.3 ETAPAS Y METODOLOGÍA	17
3.2.4 FUNDAMENTOS UTILIZADOS	21
3.2.5 IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS	36
3.3 EVALUACIÓN	76
3.3.1 EVALUACIÓN ECONÓMICA	76
	vii

CAPÍTULO IV – REFLEXIÓN CRÍTICA DE LA EXPERIENCIA	81
CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES	83
5.1 CONCLUSIONES	83
5.2 RECOMENDACIONES	85
5.3 FUENTES DE INFORMACIÓN	86
5.4 GLOSARIO	87
ANEXOS	89
ANEXO 1 – ACTA DE CONSTITUCIÓN	90
ANEXO 2 – REGISTRO DE INTERESADOS	93
ANEXO 3 – HISTORIAS DE USUARIO	94
ANEXO 4 – PRODUCT BACKLOG	96
ANEXO 5 – SPRINT BACKLOG	98
ANEXO 6 – CRONOGRAMA DEL PROYECTO	99
ANEXO 7 – PLAN DE DESPLIEGUE	100
ANEXO 8 – FORMULARIOS DE REUNIÓN	103
ANEXO 9 – DOCUMENTO DE PRUEBAS	106
ANEXO 10 – MODELADO DE DATOS	107

ÍNDICE DE FIGURAS

Figura 1. Clientes de RUQU S.A.C.	7
Figura 2. Aliados Estratégicos de RUQU S.A.C.	7
Figura 3. Organigrama de RUQU S.A.C.	9
Figura 4. Flujo de Estados	15
Figura 5. Metodología Scrum	18
Figura 6. Principios de Scrum	25
Figura 7. Framework MEANJS	33
Figura 8. Ejemplo de un documento JSON	33
Figura 9. Ejemplo de API REST con ExpressJS	34
Figura 10. Ejemplo de Implementación con AngularJS	35
Figura 11. Ejemplo de un Servidor con Node.js	36
Figura 12. Acta de Constitución del Proyecto	37
Figura 13. Interacción Quipuc, Base de Datos y el Nuevo Sistema	41
Figura 14. Inicio de Sesión en Quipuc	41
Figura 15. Acceso de Datos	42
Figura 16. Arquitectura del Proyecto	43
Figura 17. Arquitectura del Servidor	43
Figura 18. Arquitectura del Cliente	44
Figura 19. Prototipo - Vista General	44
Figura 20. Diagrama de Casos de Uso	45
Figura 21. Diagrama de Actividades: Actualizar Negocio	58
Figura 22. Diagrama de Actividades: Actualizar Documento de Pagos	58
Figura 23. Diagrama de Actividades: Generar Estado de Cuentas	59
Figura 24. Diagrama de Actividades: Notificar Negocios	59
Figura 25. Diagrama de Actividades: Activar Negocios	60
Figura 26. Diagrama de Actividades: Bloquear Negocios	60
Figura 27. Diagrama de Actividades: Enviar Recordatorios	61
Figura 28. Diagrama de Actividades: Generar Resumen de Cuenta	61
Figura 29. Diagrama de Actividades: Registrar Cuenta	62
Figura 30. Diagrama de Actividades: Aceptar Aviso de Cuenta	62
Figura 31. Diagrama de Actividades: Consulta Resumen de Cuenta	63
Figura 32. Modelo de Datos	64

Figura 33. Modelo de Datos - Colección businesses	65
Figura 34. Modelo de Datos - Colección users	65
Figura 35. Vista Login	66
Figura 36. Notificación enviada al usuario Quipuc	67
Figura 37. Negocios Notificados	68
Figura 38. Correo Electrónico con Lista de Negocios adjunto	69
Figura 39. Negocio Activado	69
Figura 40. Negocio Bloqueado	70
Figura 41. Ejemplo de Reducción de Datos	70
Figura 42. Vista General del Sistema de Administración de Cuentas	71
Figura 43. Acceso a Quipuc	72
Figura 44. Edición de estado	72
Figura 45. Nuevos Campos en Edición	73
Figura 46. Instancias de Servidores AWS EC2	75
Figura 47. Arquitectura del Servidor del Sistema de Administración de Cuentas	75
Figura 48. Costos AWS EC2	77
Figura 49. VAN vs TIR del Proyecto	80

ÍNDICE DE TABLAS

Tabla 1. Fases y Procesos de Scrum	28
Tabla 2. Product Backlog	38
Tabla 3. Sprint Backlog Resumido	40
Tabla 4. Caso de Uso: Actualizar Negocio	47
Tabla 5. Caso de Uso: Actualizar Documento de Pagos	48
Tabla 6. Caso de Uso: Generar Estado de Cuentas	49
Tabla 7. Caso de Uso: Notificar Negocios	50
Tabla 8. Caso de Uso: Activar Negocios	51
Tabla 9. Caso de Uso: Bloquear Negocios	52
Tabla 10. Caso de Uso: Enviar Recordatorios	53
Tabla 11. Caso de Uso: Generar Resumen de Cuenta	54
Tabla 12. Caso de Uso: Registrar Cuenta	55
Tabla 13. Caso de Uso: Aceptar Aviso de Cuenta	56
Tabla 14. Caso de Uso: Consultar Resumen de Cuenta	57
Tabla 15. Trazabilidad de Casos de Prueba - Requisitos	74
Tabla 16. Costos por Rol del Proyecto	76
Tabla 17. Costos por Servidor	77
Tabla 18. Resumen de Gastos	78
Tabla 19. Ingresos Estimados	78
Tabla 20. Flujo de caja proyectado	79

INTRODUCCIÓN

El presente informe de experiencia profesional describe el desarrollo de un sistema web para la administración del acceso a los usuarios que utilizan el sistema Quipuc, lo que permitirá a la empresa contar con una herramienta de gestión para sus productos.

La empresa RUQU S.A.C. se especializa en proporcionar soluciones de software a medida según las necesidades del cliente. Además, brinda un servicio para ayudar en la gestión de un negocio, publicación de productos en una tienda virtual, entre otros. La empresa se formó dentro de las aulas de la Universidad Nacional Mayor de San Marcos, y a la fecha cuenta con una amplia cartera de clientes y socios estratégicos.

Ante la necesidad de poder controlar el acceso de las cuentas el sistema Quipuc, nace la idea de implementar un Sistema de Administración de Cuentas, con el cual se podrá controlar, mediante un sistema de estados, el acceso de cada cuenta según las reglas establecidas por la empresa.

En el CAPÍTULO I se describe la experiencia del autor, los roles que ocupó y los proyectos es los que participó por empresa. Además, se listan los conocimientos técnicos, idioma extranjero y experiencias significativas que posee.

En el CAPÍTULO II se relata la historia de RUQU S.A.C. sus orígenes, su estructura orgánica, parte de sus clientes, socios estratégicos, su visión y misión y los servicios que brinda.

En el CAPÍTULO III se muestra con mayor detalle el trabajo realizado en el proyecto de Administración de Cuentas, los objetivos de haber realizado el proyecto, los procesos que se siguieron para lograrlo, conceptos utilizados y logros alcanzados.

En el CAPÍTULO IV se refiere al aporte del autor, el desarrollo profesional que le demandó, experiencia y conocimiento adquiridos.

En el CAPÍTULO V se listan las conclusiones y recomendaciones elaboradas por el autor que se pueden tener en cuenta para desarrollar este tipo de proyectos.

CAPÍTULO I – TRAYECTORIA PROFESIONAL

1.1 PRESENTACIÓN PROFESIONAL

Bachiller en Ingeniería de Software de la Universidad Nacional Mayor de San Marcos, con 4 años en experiencia como Analista Funcional en el análisis, diseño e implementación de aplicaciones web e híbridos.

A continuación un detalle de la experiencia profesional realizada:

1.2 EXPERIENCIA PROFESIONAL

EXPERIENCIA PROFESIONAL	
Analista Funcional RUQU S.A.C. Área de Proyectos Web Funciones: <ul style="list-style-type: none">- Participación del desarrollo de un Sistema de Gestión de Visitas a Practicantes para SENATI.- Integración de los sistemas ERP Navision y CRM Dynamics para la Gestión de Clientes Postulantes para la Universidad Le Cordon Bleu Perú.- Desarrollo y mantenimiento del Sistema de Gestión Administrativa para el control de pre matrículas, matrículas, notas y pagos.- Análisis para el desarrollo de la segunda versión del Sistema	Noviembre 2015 – Actualidad

<p>de Geo localización para personal en campo para TargetMaps.</p> <ul style="list-style-type: none"> - Participación en el desarrollo y mantenimiento del Sistema de Gestión de Inventario de las Tiendas Quipuc. - Mantenimiento y personalización de Tiendas Virtuales. - Desarrollo de un Sistema de Administración de Cuentas para el control del acceso a Quipuc. - Desarrollo de un Sistema de Gestión de Citas Médicas e Historial Académico para una clínica local. - Desarrollo y mantenimiento de un Sistema de Biblioteca Virtual para la Gestión de libros y tesis físico y virtual. 	
<p>Analista Programador LE CORDON BLEU PERÚ Área de Sistemas</p> <p>Funciones:</p> <ul style="list-style-type: none"> - Participación en el desarrollo y mantenimiento del Sistema de Gestión Administrativa para el control de pre matrículas, matrículas, notas y pagos. - Desarrollo y mantenimiento de la Intranet para Alumnos, Docentes y Empresas Externas, para el manejo de matrículas, horarios, cuenta corriente, control de asistencias de alumnos y docentes, notas y bolsas de trabajo. - Participación en la integración del Sistema ERP Navision con el Sistema de Gestión Administrativa para el manejo de pagos, materiales y envíos de cuotas por pagar al banco. 	<p>Julio 2014 – Noviembre 2015</p>
<p>Analista Programador UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS Facultad de Ingeniería de Sistemas e Informática Departamento de Desarrollo</p> <p>Funciones:</p> <ul style="list-style-type: none"> - Participación del desarrollo y mantenimiento del Sistema Biométrico de Asistencia Docente. - Desarrollo del Sistema de Control de Asistencias Docente. 	<p>Noviembre 2013 – Julio 2014</p>

- Desarrollo del Sistema Biométrico de Asistencia a Asambleas para el control de quórum para asambleas.	
---	--

1.3 FORMACIÓN PROFESIONAL

FORMACIÓN PROFESIONAL	
Grado Académico de Bachiller en Ingeniería de Software Escuela Académico Profesional de Ingeniería de Software Facultad de Ingeniería de Sistemas e Informática Universidad Nacional Mayor de San Marcos	Marzo 2010 – Diciembre 2014

1.4 IDIOMAS

IDIOMAS	
Inglés Universidad Nacional Mayor de San Marcos Curso Básico / Intermedio Pregrado	Marzo 2010 – Diciembre 2012

1.5 OTROS CONOCIMIENTOS

OTROS CONOCIMIENTOS	
Lenguajes de Programación	Java, JavaScript, TypeScript.
Frameworks	Primefaces, Spring, Hibernate, MEANJs, AngularJs, Bootstrap.
SVC	Git, SVN.
Bases de Datos	MySQL, PostgreSQL, SQL Server, MongoDB.
Gestores de Base de Datos	MySQL Workbench, Pg Admin, Microsoft SQL Server Management Studio, RoboMongo.
Metodologías	SCRUM, RUP, PMI, PMBOK.
Herramientas de Gestión	JIRA, Microsoft Project.
Herramientas de Modelado	Rational Rose, Bizagi, Power Designer.
Herramientas de Diseño de Prototipos	Balsamiq Mockups

CAPÍTULO II – CONTEXTO EN EL QUE SE DESARROLLÓ LA EXPERIENCIA

2.1 EMPRESA - ACTIVIDAD QUE REALIZA

RUQU es una consultora de software, que nace de la unión de cuatro compañeros pertenecientes a la Facultad de Ingeniería de Sistemas e Informática de la Universidad Nacional Mayor de San Marcos.

Luego de elaborar el proyecto de un curso de pregrado que cursaban, deciden continuar con dicho proyecto naciendo así la sociedad RUQU, el cual inicia sus operaciones formalmente en el año 2013. Más adelante RUQU adopta el nombre comercial “DMT Technology” para darse a conocer dentro del ámbito del desarrollo de software.

Datos de la empresa:

- Razón Social: RUQU S.A.C.
- R.U.C.: 20552993404
- Domicilio legal: Jr. Julio Rodríguez Nro. 373 U. Pop Ciudad de Dios (Alt. de Pista Nueva con Av. San Juan de Miraflores), San Juan de Miraflores, Lima, Perú.
- Actividad Económica: Actividades de Tecnología de la Información y de Servicios Informáticos
- Fundación: 2013
- Teléfono: (01) 276 3882
- Página web: <http://www.dmt.pe/>

Como parte de la cartera de clientes que maneja, a quienes se les brinda el servicio de mantenimiento e implementación de nuevas características para sus páginas o sistemas web, se tiene a las siguientes empresas:

- Le Cordon Bleu Perú
- SENATI
- Target Maps
- Laboratorios SMASAC
- IMADIC Marketing Digital
- Austral Corredores de Seguros
- Activa Ambientes Inteligentes.

Figura 1

Clientes de RUQU S.A.C.



Fuente: (RUQU, 2018)

Figura 2

Aliados Estratégicos de RUQU S.A.C.



Fuente: (RUQU, 2018)

La empresa cuenta con una amplia lista de aliados estratégicos, con los que realiza en conjunto proyectos de mediana y gran envergadura, permitiendo administrar mejor los recursos internos de la empresa. Entre ellas tenemos a:

- DISNOVO
- AWS Perú
- Cámara de Comercio de Lima.
- Ruta Estratégica Empresarial.
- Branding para Pymes
- Tusán Business
- Arco Brands Consultant
- EK Consulting Group

RUQU S.A.C. brinda servicios enfocados a:

Fábrica de Software:

Utilizando lo último de metodologías de desarrollo de software, tecnologías y buenas prácticas, se implementan soluciones de software según la necesidad del negocio y del mercado.

Outsourcing de sistemas y aplicaciones:

Brinda servicios de asesoramiento, diseño, gestión de proyectos de software inhouse, aportando la experiencia ganada a lo largo de diferentes proyectos ya implementados, utilizando buenas prácticas y herramientas que cubran las necesidades del cliente. De este modo, el cliente puede enfocar mayor atención a controlar su propio negocio.

Soluciones TI:

Desarrolla soluciones tecnológicas, aplicaciones, integración de diversos sistemas mediante el uso de metodologías que aseguren la calidad del producto en base a las necesidades planteadas por el cliente.

2.2 VISIÓN

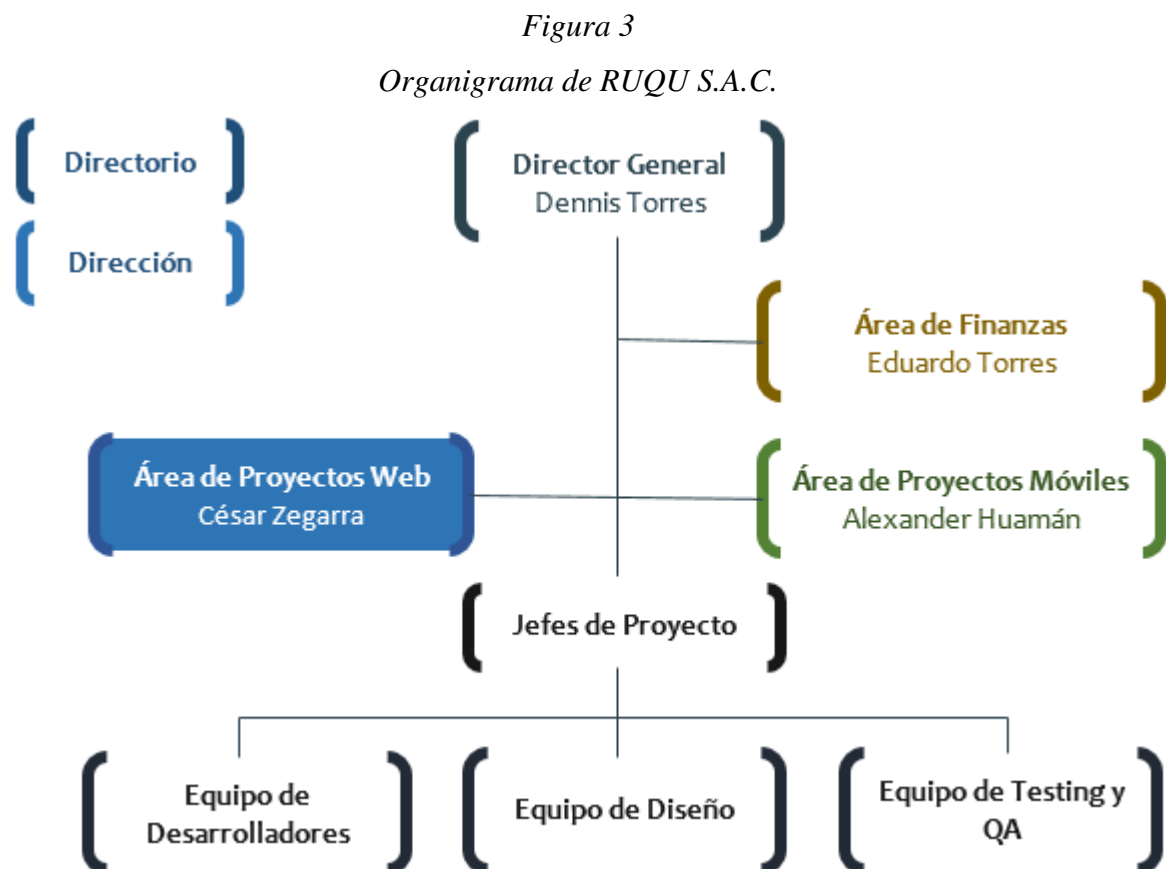
“Encabezar el mercado de soluciones de software y TI como una empresa que tenga como fortaleza la creatividad, innovación, mejora continua y valorización del talento de las personas.” (RUQU, 2018)

2.3 MISIÓN

“Creemos en que siempre hay algo por mejorar, seguir innovando y creciendo es parte de nuestro ADN. Además, sabemos que esto permite seguir mejorando para brindar el mejor servicio con la mejor tecnología para acelerar su proyecto.” (RUQU, 2018)

2.4 ORGANIZACIÓN DE LA EMPRESA

RUQU S.A.C. es una empresa con una estructura organizacional orientada a proyectos. El siguiente organigrama corresponde al período en el que se desarrolló el Sistema de Administración de Cuentas:



Fuente: (Dirección RUQU, 2018)

De la Figura 3 destacamos las dos grandes áreas: El Área de Proyectos Web y el Área de Proyectos Móviles. Éstas dos áreas están encabezadas por un jefe de área y tienen a disposición al personal que requieran (jefes de proyecto, desarrolladores, etc.) según el proyecto que se necesite realizar, necesite más personal o se reasigne, según lo evaluado y coordinado entre los jefes de área y el director general.

2.5 ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS

2.5.1 Área

El área donde se desarrolló la experiencia profesional fue el Área de Proyectos Web.

2.5.2 Cargo

El autor de este informe de experiencia profesional se desempeñó como Analista Funcional, con participación en proyectos del tipo Backend y Frontend, bajo el Área de Proyectos Web de la empresa en enero del 2017.

2.5.3 Funciones desempeñadas

Dentro de las funciones desempeñadas estuvo el levantar requerimientos, desarrollar las historias de usuarios, plantear la arquitectura del proyecto, codificación, despliegue y soporte ante cualquier problema presente. La comunicación y reuniones con el product owner fueron semanales, desde el levantamiento de requerimientos hasta la capacitación para utilizar el sistema desarrollado.

2.6 EXPERIENCIA PROFESIONAL REALIZADA EN LA ORGANIZACIÓN

Proyecto: QUIPUC

Quipuc es un proyecto que elabora un aplicativo en JavaScript capaz de ser desplegado tanto en una plataforma web como en una plataforma móvil Android mediante Google Play, lo que se podría llamar como aplicativo híbrido. El aplicativo tiene por finalidad administrar el inventario de un negocio, desde el manejo de los productos, stocks, precios, proveedores, clientes, compras y ventas; además, dispone de varios reportes útiles para que el usuario pueda controlar mejor los activos de su negocio.

Principales tareas realizadas en el proyecto:

- Levantamiento de requerimientos para cambios, nuevas funcionalidades y/o soporte ante la aparición de defectos.
- Publicación de una nueva versión del producto.
- Reuniones con el product owner.

Proyecto: QUIPUC ADMIN

Este proyecto nace en base a la necesidad de controlar el acceso de las cuentas Quipuc, dicho control se logra gracias a la implementación de un “Sistema de Estados”. Cada negocio posee un estado que irá cambiando según el día del mes o si se cancela alguna deuda pendiente que tenga el negocio con la empresa. El aplicativo está desarrollado en JavaScript y entre sus principales funciones tiene las de generar y enviar notificaciones, realizar el cálculo automático del monto a cobrar por concepto de uso y cambio de estados a las cuentas de la aplicación Quipuc.

Principales tareas ejecutadas en el proyecto:

- Levantamiento de requerimientos para el desarrollo del proyecto.
- Especificación de requerimientos.
- Planteamiento de la arquitectura.
- Desarrollo y corrección de defectos encontrados.
- Publicación de una nueva versión del producto.
- Reuniones con el product owner.

Proyecto: SISBIB

El proyecto está desarrollado en Java Web y JSF, y se encarga de administrar la biblioteca de la Universidad Nacional José Faustino Sánchez Carrión. El sistema, como parte del control de la biblioteca, administra el stock de los libros físicos de la universidad, verifica su disponibilidad, muestra al usuario a quien se le prestó un ejemplar y por cuánto tiempo lo va a utilizar, etc. Además, permite subir libros virtuales o tesis aprobados digitalizados en formato PDF.

Principales tareas realizadas en el proyecto:

- Levantamiento de requerimientos para el desarrollo del proyecto.
- Especificación de requerimientos.
- Planteamiento de la arquitectura.

- Desarrollo y corrección de defectos encontrados.

Proyecto: SISMEDIC

SISMEDIC es un sistema que controla el historial clínico de un paciente y las citas que este tiene mediante un calendario de citas. Además, realiza un control de los tratamientos a realizar según la enfermedad y/o malestar presente en el paciente en base al CIE (Código Internacional de Enfermedades) diagnosticado al paciente.

Principales tareas ejecutadas en el proyecto:

- Levantamiento de requerimientos para el desarrollo del proyecto.
- Especificación de requerimientos.
- Planteamiento de la arquitectura.
- Desarrollo y corrección de defectos encontrados.

Proyecto: Easybuy Express

Easybuy Express es una empresa que se dedica a la venta de artículos variados mediante su tienda virtual. Se encarga de mostrar, cómo en vitrina, los productos para el hogar, para el vehículo, etc. Además ofrece la posibilidad de separar stock y comprar, mediante una pasarela de pagos, con una tarjeta de crédito / débito, contra entrega, depósito o recojo en tienda.

Principales tareas realizadas en el proyecto:

- Levantamiento de requerimientos para el desarrollo del proyecto.
- Especificación de requerimientos.
- Planteamiento de la arquitectura.
- Desarrollo y corrección de defectos encontrados.

Proyecto: QUIPUC API

Quipuc API nace en base a la necesidad de desarrollar y desplegar tiendas virtuales personalizadas por negocio. Al ser genérico se optó por un tipo de servicio bien conocida por la empresa que sería un API RESTful, con métodos GET y POST que permitirían consultar los datos de los productos y clientes del aplicativo Quipuc, teniendo la posibilidad de realizar compras mediante una pasarela de pagos integrada en el sistema.

Principales tareas ejecutadas en el proyecto:

- Desarrollo y corrección de defectos encontrados.
- Publicación de una nueva versión del producto.

Proyecto: SENATI

El proyecto administra las visitas que realizan los instructores a cada alumno que esté a su cargo. Cada visita es programada y de esta se elaboran formularios que deben ser completados por los instructores en base a criterios, los criterios son evaluados según la labor que realice el alumno en la empresa en el cual esté realizando sus prácticas pre profesionales. El proyecto también permite postular a un alumno a una empresa que requiera personal para que realice sus prácticas pre profesionales.

Principales tareas realizadas en el proyecto:

- Levantamiento de requerimientos para el desarrollo del proyecto.
- Especificación de requerimientos.
- Planteamiento de la arquitectura.
- Desarrollo y corrección de defectos encontrados.

Proyecto: SGA

El proyecto SGA (Sistema de Gestión Académica) es planteada como reemplazo de un sistema anterior que estaba teniendo problemas en soportar las nuevas exigencias de la institución Le Cordon Bleu Perú que administra una Universidad e Instituto, por ende se plantea desarrollar una aplicación web para administrar todas las actividades académicas, que se pueda integrar con otras herramientas externas (ERP, CRM, etc) para mejorar el control en la institución. Como labores principales administra los períodos académicos, carreras u ofertas académicas, cursos o demostraciones, postulantes o alumnos, cuotas, notas, material de clase, asistencias de docentes y alumnos, pre matrículas, matrículas, entre otros.

Principales tareas ejecutadas en el proyecto:

- Especificación de nuevos requerimientos y estimaciones.
- Desarrollo y corrección de defectos encontrados.
- Publicación de una nueva versión del producto.
- Reuniones con el cliente.

CAPÍTULO III – ACTIVIDADES DESARROLLADAS

3.1 SITUACIÓN PROBLEMÁTICA

3.1.1 DEFINICIÓN DEL PROBLEMA

El sistema Quipuc, en adelante Quipuc, es un aplicativo híbrido que permite administrar el inventario de un determinado negocio. Cada cuenta Quipuc se basa en un único negocio, que tiene de un usuario a más.

Quipuc se publicó con un acceso básico de usuario – contraseña, permitiendo que los usuarios puedan acceder libremente simplemente creando una cuenta, ocasionando esto que no se tenga una administración en el acceso de dichos usuarios en lo que se refiere a tiempo y costo de uso, cantidad de usuarios, notificaciones y control de estados.

Existen negocios que usan el sistema Quipuc, con diferente cantidad de usuarios, muchos con un único usuario, algunos con 2 y otros de 3 a más. El costo de uso mensual por un usuario es de S/ 50 (S/ 59 incluyendo IGV).

Entonces, al no contar con un sistema que administre el acceso a Quipuc, y tener registrado más de 1300 negocios, esto significa una pérdida económica mensual considerable para la empresa.

3.2 SOLUCIÓN

Analizando la problemática, se propuso desarrollar un sistema que permita administrar las cuentas de acceso a Quipuc. Con mejor tecnología y mejor gestionado.

Se planteó implementar un sistema de estados, donde a la cuenta se le asignará un estado y, según la ejecución de los procesos del nuevo sistema, la cuenta cambiará de estado durante el transcurso del mes.

Figura 4
Flujo de Estados



Fuente: Elaboración Propia.

El estado “**gratuito**” no entra en el flujo de estados debido a que no genera ninguna notificación ni cuenta por pagar, y cuenta con acceso libre e indefinido.

El estado “**ninguno**” (también llamado como “**sin estado**”) se le asigna a una cuenta nueva, tiene una cantidad de días de prueba sin costo, y entra al flujo automático sólo si el usuario dueño de la cuenta acepta un compromiso para usar el sistema.

El estado “**bloqueado**” restringe por completo el acceso a la cuenta, y no podrá acceder nuevamente hasta que realice el pago generado por uso del sistema.

Las cuentas Quipuc con estado **“notificado”** o **“activo”** tienen acceso al sistema hasta que el nuevo sistema los cambie mediante procesos automáticos.

Los cambios de estado se dan en fechas específicas mediante la ejecución de procesos automáticos. Cada proceso debe actualizar el estado de las cuentas Quipuc según el estado actual en el que se encuentre, dejando las otras cuentas, que no tengan asignado dicho estado, sin alterar.

Además, es necesario un control manual sobre los estados. Esto en caso se necesite modificar el estado de la cuenta Quipuc, así como otra información adicional que tenga la cuenta.

Se propuso, adicional a lo anterior, que un administrador (usuario del nuevo sistema) pueda acceder a una cuenta Quipuc y pueda otorgar asistencia en caso el cliente, dueño de la cuenta, requiera algún tipo de soporte o ayuda con su cuenta.

3.2.1 OBJETIVOS

OBJETIVO GENERAL

Desarrollar un Sistema de Administración de Cuentas para mejorar el control de accesos al Sistema Quipuc de una Empresa de Consultoría de Software.

OBJETIVOS ESPECÍFICOS

- Crear procesos automatizados para la actualización de los estados de las cuentas según el día del mes (inicio, quincena, fin de mes).
- Notificar a la cuenta el costo por suscripción generado por el uso del sistema Quipuc.
- Crear un acceso para la ejecución manual de los procesos automatizados.
- Permitir el control continuo de los estados de las cuentas de usuario.
- Permitir acceder a una cuenta Quipuc desde el sistema para brindar soporte al cliente dueño de dicha cuenta.

3.2.2 ALCANCE

ALCANCE FUNCIONAL

El alcance de funcionalidades del nuevo proyecto se basa en las historias de usuario recopiladas del Product Owner (ver Anexo 3), de ellas se analizó los requerimientos principales y según acuerdos con el Product Owner se elaboró una lista de funcionalidades final.

ALCANCE TECNOLÓGICO

El proyecto a desarrollar se podrá utilizar en navegadores web como Google Chrome versión 40 o superior, Mozilla Firefox versión 48 o superior o Microsoft Edge versión 30 o superior.

Para el desarrollo del proyecto se utilizará Node.js 6.9.2 como plataforma de desarrollo en la parte del servidor (Backend) y AngularJs 1 para la parte del cliente (Frontend).

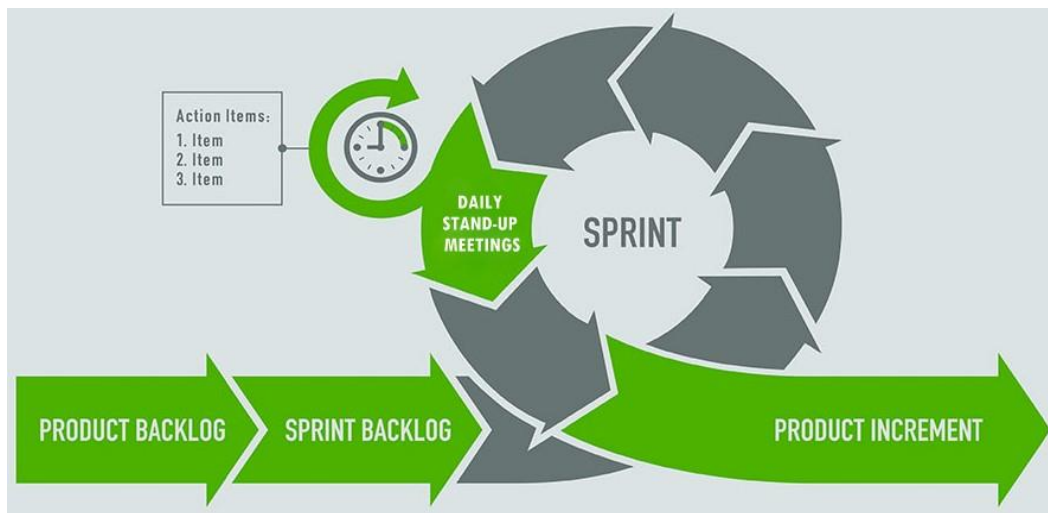
ALCANCE ORGANIZACIONAL

El proyecto involucra, dentro de la empresa, al Director General como dueño del producto o “Product Owner” y al Área de Proyectos Web al que pertenece el equipo que realizará el desarrollo del proyecto.

3.2.3 ETAPAS Y METODOLOGÍA

El proyecto se desarrolló en 6 etapas: Inicio, Planificación y Requerimientos, Análisis y Diseño, Implementación, Pruebas y por último el Despliegue y Mantenimiento. Todas estas etapas están basadas en la Metodología Ágil Scrum que permite el desarrollo continuo e incremental del proyecto en base a sus iteraciones o sprints, que son intervalos de tiempos fijos en los que se asignan tareas al equipo para desarrollarlas y presentarlas al Product Owner como informe de avance cuando las iteraciones o sprints terminen (ver Figura 5).

Figura 5
Metodología Scrum



Fuente: (SCRUMstudy, 2018)

INICIO

Etapa inicial del proyecto en el cual, mediante una serie de reuniones entre el Product Owner, el Scrum Master y, en ciertas ocasiones, un elemento del equipo, se planea definir lo siguiente:

- Alcance del proyecto.
- Scrum Master del proyecto.
- Equipo de trabajo.

Lo roles involucrados en esta etapa del proyecto son:

- Product Owner.
- Scrum Master.
- Analista Funcional.

Los entregables que se esperan generar en esta etapa del proyecto son:

- Acta de Constitución del proyecto (ver Anexo 1).
- Historias de Usuario (ver Anexo 3).

PLANIFICACIÓN Y REQUERIMIENTOS

Siguiente etapa del proyecto en el cual se planifica lo que se va a realizar en el proyecto en base al alcance definido para el proyecto como:

- Tiempo estimado para el proyecto.
- Número de sprints y el tiempo de su duración.
- Estimación y confirmación de las historias de usuario.
- Plan de reuniones con el Product Owner.
- Priorización del Product Backlog.
- Plan de Despliegue y Pruebas.

Lo roles involucrados en esta etapa del proyecto son:

- Product Owner.
- Tester.
- Scrum Master.
- Analista Funcional.

Los entregables que se esperan generar en esta etapa del proyecto son:

- Product Backlog (ver Anexo 4).
- Sprint Backlog (ver Anexo 5).
- Cronograma del Proyecto (ver Anexo 6).
- Plan de Despliegue (ver Anexo 7).
- Documento de Pruebas (ver Anexo 9).

ANÁLISIS Y DISEÑO

Tercera etapa del proyecto, posiblemente el más corto pero no menos importante del proyecto. En esta etapa se definen elementos básicos para que el proyecto plante sus cimientos e inicie su desarrollo como:

- Arquitectura a utilizar.
- Prototipos.

Lo roles involucrados en esta etapa del proyecto son:

- Scrum Master.
- Analista Funcional.

Los entregables que se esperan generar en esta etapa del proyecto son:

- Arquitectura del Proyecto.
- Prototipo - Vista General.

IMPLEMENTACIÓN

Cuarta etapa del proyecto, en esta etapa se realiza el desarrollo del proyecto según los sprints definidos en la etapa de planificación, el principal interés es cumplir con siguiente.

- Desarrollo de las tareas definidas en el Sprint Backlog.
- Tener reuniones con el Product Owner para mostrar el avance.

Lo roles involucrados en esta etapa del proyecto son:

- Product Owner.
- Scrum Master.
- Analista Funcional.
- Analista Programador.

Los entregables que se esperan generar en esta etapa del proyecto son:

- Software de Administración de Cuentas.
- Formularios de Reunión (ver Anexo 8).

PRUEBAS

Penúltima etapa del proyecto se el cual se ejecuta el documento de pruebas que fue elaborado previamente, según los resultados del mismo se envían las observaciones para ser subsanadas por el equipo de desarrollo.

Roles involucrados:

- Scrum Master.
- Analista Funcional.
- Tester.

Documentación generada:

- Documento de pruebas (ver Anexo 9).

DESPLIEGUE Y MANTENIMIENTO

Última etapa del proyecto donde se pone el sistema desarrollado a disposición completa del Product Owner y de los usuarios finales. Además se elabora un documento que recopila las lecciones aprendidas en el presente proyecto.

Roles involucrados:

- Product Owner
- Scrum Master.
- Analista Funcional.

Documentación generada:

- Plan de Despliegue (ver Anexo 7).
- Lecciones Aprendidas.

Las 6 etapas deben ser superadas por el equipo de trabajo, liderados por el Scrum Master, que deberá cumplir con los tiempos establecidos y entregar un producto acorde con la necesidad del Product Owner.

En la sección 3.2.5 Implementación de las áreas, procesos, sistemas y buenas prácticas, se mostrará con mayor detalle el proceso realizado y los entregables de cada etapa.

3.2.4 FUNDAMENTOS UTILIZADOS

PROYECTO SEGÚN EL PMI

Un proyecto es una actividad en equipo con el fin de producir un producto, servicio, o resultado único (PMIlatam, 2018).

Es temporal dado que se establece un inicio y un fin y por lo tanto un alcance, estos pueden variar según varios factores, pero siempre cuentan con recursos definidos (PMIlatam, 2018)..

Es único ya que es un conjunto específico de operaciones diseñadas para lograr una meta particular, por ende deja de una operación rutinaria. Un equipo de proyecto se conforma por personas que no necesariamente trabajan

juntas, y en ocasiones la conforman elementos de otras organizaciones, regiones o incluso países (PMIlatam, 2018)..

Entre los diversos ejemplos de proyectos tenemos el desarrollo de un software para mejorar un proceso de negocio, la construcción de un puente o de un edificio, un esfuerzo de recuperación luego de un desastre natural, o la entrada a un nuevo mercado.

GESTIÓN DE PROYECTOS SEGÚN EL PMI

La gestión de proyectos implica aplicar el conocimiento, las habilidades y técnicas adquiridas por el gestor para ejecutar los proyectos que se le encarguen de manera eficiente y efectiva. Las empresas la utilizan estratégicamente para obtener resultados que le ayuden a cumplir metas del negocio, y así competir mejor en su mercado (PMIlatam, 2018)..

De la unión de las buenas prácticas y lecciones aprendidas a lo largo de diferentes proyectos, comienza a La Guía de los Fundamentos para la Gestión de Proyectos (Guía del PMBOK), y se identifican sus elementos recurrentes:

Los procesos para dirigir los proyectos caen en cinco grupos o etapas:

- Iniciación
- Planificación
- Ejecución
- Monitoreo y Control
- Cierre

Áreas de conocimiento de la dirección de proyectos:

- Gestión de la integración
- Gestión del alcance
- Gestión del tiempo
- Gestión del costo
- Gestión de la calidad
- Gestión de adquisiciones
- Gestión de recursos humanos
- Gestión de las comunicaciones

- Gestión de riesgos
- Gestión de los interesados

La dirección de proyectos se enfoca en el cumplimiento de las metas, el uso adecuado de los recursos y la elaboración del cronograma de cada proyecto. Dicho enfoque ha resultado ser muy exitoso por el crecimiento rápido que ha tenido por todo el mundo:

- Como una competencia estratégica de la organización,
- Como un tema para la capacitación y la educación, y
- Como un camino para el desarrollo profesional y de carrera.

METODOLOGÍA SCRUM

“Es una de las metodologías ágiles más populares. Es una metodología adaptable, iterativa, rápida, flexible y efectiva, diseñada para entregar un valor significativo rápidamente y durante un proyecto. Scrum garantiza la transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y progreso continuo. El marco Scrum, tal como se define en la Guía SBOK, está estructurado de tal forma que admite el desarrollo de productos y servicios en todo tipo de industrias y en cualquier tipo de proyecto, independientemente de su complejidad” (SCRUMstudy, 2018).

La principal fortaleza de Scrum se basa en la conformación de equipos cuyos miembros puedan tener diferentes especialidades y auto organizados que puedan dividirse el trabajo de cada Sprint (ciclos cortos de trabajo).

“El ciclo de Scrum comienza con una reunión de partes interesadas, durante la cual se crea la visión del proyecto. Luego, el Propietario del producto desarrolla una Cartera priorizada de productos que contiene una lista priorizada de los requisitos comerciales y del proyecto escritos en forma de Historias de usuarios. Cada Sprint comienza con una Reunión de planificación de Sprint durante la cual se consideran las Historias de usuarios de alta prioridad para su inclusión en Sprint. Un Sprint generalmente dura entre una y seis semanas e involucra al equipo de Scrum que trabaja para crear entregas potencialmente entregables o incrementos de productos. Durante el Sprint, se llevan a cabo Reuniones de Standup breves, altamente enfocadas,

donde los miembros del equipo analizan el progreso diario. Hacia el final del Sprint, se lleva a cabo una reunión de revisión de Sprint durante la cual el propietario del producto y las partes interesadas relevantes reciben una demostración de los entregables. El propietario del producto acepta los entregables solo si cumplen los criterios de aceptación predefinidos. El ciclo de Sprint finaliza con una reunión de Retrospect Sprint donde el equipo analiza formas de mejorar los procesos y el rendimiento a medida que avanzan en el siguiente Sprint” (SCRUMorg, 2018).

Principios de Scrum

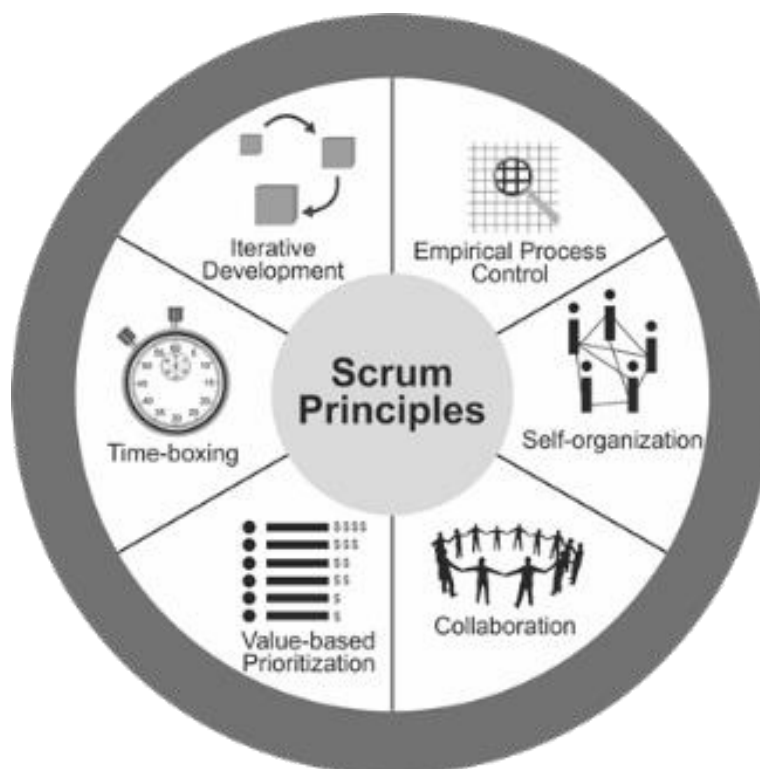
Los principios de Scrum son las pautas principales para aplicar el marco de Scrum y deben ser obligatoriamente utilizados en todos los proyectos de Scrum. Como se especifica en el SBOK estos principios no son negociables y deben aplicarse tal cual. Con esto se logra infundir confianza en el marco de Scrum respecto al logro de los objetivos del proyecto.

Los seis principios de Scrum (SCRUMstudy, 2018):

- **Control de proceso empírico:** Basada en las tres ideas principales de transparencia, inspección y adaptación, filosofía central que enfatiza Scrum.
- **Auto organización:** Centrado en los colaboradores que tiene la organización, cuyo valor crece significativamente cuando se auto organizan y lo que genera una mejor aceptación del equipo y una propiedad compartida, además de un entorno innovador y creativo que propicia el crecimiento.
- **Colaboración:** Aboga por la gestión de proyectos como un proceso que crea valor compartido entre equipos que colaboran entre sí para llegar a ofrecer mayor valor dentro de la organización. Son tres las dimensiones centrales en las que se centra este principio respecto al trabajo colaborativo: conciencia, articulación y apropiación.

- **Priorización basada en el valor:** Se basa en ofrecer el máximo valor comercial desde que el proyecto inicia, durante la implementación hasta la finalización del mismo.
- **Time-boxing:** El tiempo siempre será considerado un recurso limitante en Scrum por lo que este principio es utilizado para mejorar la administración, planificación y ejecución del proyecto. Los Sprints, Reuniones de planificación y revisión de Sprint y otros que signifiquen el uso del recurso “tiempo” son elementos que Scrum incluye en el Time-boxed.
- **Desarrollo iterativo:** Se podría resumir en “Desarrollo y Error”. Enfatiza el desarrollo iterativo con gestión de cambios en base a las reuniones que se realicen durante la construcción del proyecto, logrando crear así productos que lleguen a satisfacer las necesidades de los stakeholders.

Figura 6
Principios de Scrum



Fuente: (SCRUMstudy, 2018)

Aspectos de Scrum

Los aspectos de Scrum deben abordarse y administrarse a lo largo de un proyecto de Scrum como se explica en SBOK. Los cinco aspectos de Scrum son (SCRUMstudy, 2018):

- **Organización:** Aspecto que se centra en varias facetas de una organización de proyecto de Scrum, roles básicos y no básicos, y cómo establecer equipos Scrum con buen rendimiento.
- **Justificación comercial:** Aspecto que se centra en el concepto y el objetivo de la Justificación empresarial en lo que respecta a los proyectos de Scrum.
- **Calidad:** Aspecto que se centra en definir la calidad en relación con los proyectos, presentando el enfoque de Scrum para obtener la mayor calidad que se requiere.
- **Cambio:** Aspecto que se centra en la importancia del cambio en cualquier proyecto, sea cual sea su metodología o marco de trabajo, y se expande sobre cómo los procesos de desarrollo de Scrum están diseñados para abarcar el cambio.
- **Riesgo:** Aspecto que se centra en la gestión de riesgos de un proyecto Scrum considerando múltiples herramientas que ayudan a facilitar la gestión de riesgos de la mejor manera.

Roles Principales en Scrum (SCRUMstudy, 2018)

- **Product Owner:** Es el encargado de que el equipo “Scrum” trabaje bajo la perspectiva del negocio. Este rol también se encarga de apoyar al usuario en la elaboración de las historias de usuario, priorizarlas y finalmente elaborar el Product Backlog.
- **ScrumMaster (o Facilitador):** Se encarga de facilitar la ejecución del Scrum, debiendo eliminar cualquier impedimento o incidencia que afecte al equipo para así lograr completar satisfactoriamente,

o en su mayoría, el sprint. Este rol debe actuar como una barrera entre el equipo y las influencias externas que afecten la normal ejecución de las tareas. Además, debe asegurar que el proceso Scrum sea realizada correctamente, haciendo cumplir las reglas como es debido.

- **Equipo de desarrollo:** Este “rol” tiene por finalidad la elaboración o implementación de la solución o producto. Su composición puede ser de 5 a 9 integrantes que tengan habilidades transversales para que el análisis, diseño, desarrollo, pruebas, documentación, etc. sean llevadas a cabo de la mejor manera.

Roles Auxiliares

Son roles no formales y suelen no involucrarse en el “Proceso Scrum”, pero su opinión deben ser tomada en cuenta ya que, como metodología ágil, es importante involucrar en el proceso a los usuarios, expertos del negocio y otros interesados (Stakeholders). Con la finalidad de que realicen revisiones y den una retroalimentación del entregable del sprint y poder planear el siguiente.

- **Stakeholders:** Son las personas “interesadas” en la realización de proyecto (clientes, proveedores, vendedores, etc.) ya que este les traerá un posible beneficio, justificando así el desarrollo del mismo. Suelen participar durante las revisiones del “Sprint”.
- **Administradores (Managers):** Se encargan de la gestión de los recursos para el desarrollo del proyecto.

Fases y procesos de Scrum (SCRUMstudy, 2018)

Según la Guía SBOK, existen un total de 19 procesos que se agrupan en 5 fases: “Inicio”, “Planificación y Estimación”, “Implementación”, “Revisión y Retrospectiva” y “Despliegue”, que se explican a continuación:

1. **Inicio:** Esta fase, como parte inicial del proyecto, contempla los siguientes procesos: Crear Visión del proyecto, Identificar Scrum

Master y Stakeholders, Formar el Equipo Scrum, Desarrollar Épicas, Priorizar el Product Backlog y Plan de Despliegue.

2. **Planificación y Estimación:** Esta fase contempla procesos de análisis, planificación de tareas y su respectiva estimación como: Crear las Historias de Usuario, Aprobar, Estimar y Confirmar las Historias de Usuario, Crear tareas, Estimar tareas y Crear el Sprint Backlog.
3. **Implementación:** Esta fase, relacionada con la ejecución, contempla procesos para la creación del producto de un proyecto tales como: Creación de los distintos entregables, Realizar de reuniones diarias y Revisión y Actualización periódica del Product Backlog a intervalos regulares.
4. **Revisión y Retrospectiva:** Esta fase contempla procesos de revisión. Se revisa el entregable obtenido de la implementación del proyecto, Formas de mejorar las prácticas y los métodos utilizados para realizar el trabajo del proyecto.
5. **Despliegue:** Esta fase contempla los procesos finales, como: Entrega del producto aceptado al cliente, así como la elaboración de las lecciones aprendidas surgidas durante el proyecto.

Tabla 1

Fases y Procesos de Scrum

Fase	Procesos
Inicio	<ol style="list-style-type: none"> 1. Crear visión del proyecto. 2. Identificar el Scrum Master y los Stakeholders. 3. Formar el Equipo Scrum. 4. Desarrollar las Épicas. 5. Priorizar el Product Backlog. 6. Realizar el Plan Despliegue.
Planificación y Estimación	<ol style="list-style-type: none"> 7. Crear las Historias de Usuario 8. Aprobar, estimar y confirmar las Historias de Usuario. 9. Crear tareas. 10. Estimar tareas. 11. Crear el Sprint Backlog.

Implementación	12. Crear entregables. 13. Llevar a cabo presentaciones diarias. 14. Revisar y actualizar periódicamente el Product Backlog.
Revisión y Retrospectiva	15. Convocar el Scrum de Scrums 16. Demostrar y Validar el Sprint 17. Realizar una retrospectiva del Sprint.
Despliegue	18. Entregar los Entregables. 19. Realizar una retrospectiva del Proyecto.

Fuente: (SCRUMstudy, 2018)

SERVICIOS RESTful

REST son las siglas de “Representational State Transfer”, y fue descrito por primera vez dentro de una tesis en el año 2000 por Roy Fielding, considerado como padre de la especificación HTTP.

Un REST service es un conjunto de restricciones con las que se puede crear un estilo de arquitectura software, por lo que suele confundir como una. El servicio puede ser utilizado para crear aplicaciones web respetando HTTP (Idento, 2018).

Por ser un estándar muy eficiente, hoy en día muchos de los servicios implementados por las empresas se hacen basados en el API REST. Algunos ejemplos que se encuentran hoy en día en internet son los sistemas de identificación de Facebook, Suite de Google (Drive, Analytics, etc.) (Idento, 2018).

A continuación algunas restricciones que, según Fielding, definen a un sistema RESTful (Idento, 2018):

- **Cliente - Servidor:** Con esta restricción se logra mantener al cliente débilmente acoplado con el servidor. En otras palabras el “cliente” realizará peticiones sin importarle como el “servidor” resuelva las mismas, y el “servidor” entregará los resultados sin importarle como el “cliente” las use.

- **Sin estado:** Con esta restricción se logra que las peticiones “cliente” – “servidor” sean independientes, sin necesidad de una “sesión” activa entre ambos.
- **Cacheable:** Es posible que varias peticiones de un mismo “cliente” quieran recuperar un mismo recurso, lo que generará muchas conexiones innecesarias. Por lo que se debe manejar un sistema de almacenamiento en caché que evitará dichas conexiones.
- **Interfaz uniforme:** Esta restricción indica que cada recurso debe tener una única dirección “URI” en el servicio REST. Con ello se logra que cada interacción “cliente” – “servidor” que se produzca sea de manera uniforme, simplificando así la arquitectura haciéndola más entendible.
- **Sistema de capas:** Para ayudar en la escalabilidad, rendimiento, acoplamiento y seguridad, el “servidor” puede estar compuesto por múltiples capas.

Métodos HTTP para Servicios RESTful

Los métodos HTTP constituyen una parte importante de nuestra restricción de "Interfaz Uniforme" y nos proporcionan la contraparte de acción del recurso basado en sustantivos. Los métodos HTTP más utilizados son POST, GET, PUT, PATCH y DELETE. Estos corresponden a operaciones de creación, lectura, actualización y eliminación (o CRUD) respectivamente. También hay otros métodos, pero se utilizan con menos frecuencia como por ejemplo: OPTIONS y HEAD (RESTAPItutotial, 2018).

- **GET:** El método GET se usa para “leer” la representación de un recurso. Cuando el método se ejecuta sin errores devuelve una representación en XML o JSON y un código de respuesta HTTP de 200 (OK). Según el diseño de la especificación HTTP, las solicitudes GET (junto con HEAD) se utilizan solo para leer datos y no para modificarlos. Por lo tanto, cuando se usan de esta manera, se consideran seguros. Es decir, se pueden invocar sin riesgo de

modificación o corrupción de datos. Llamarlo una vez tiene el mismo efecto que llamarlo 10 veces o ninguno. Además, GET (y HEAD) es idempotente, lo que significa que hacer múltiples solicitudes idénticas termina teniendo el mismo resultado que una sola solicitud.

- **POST:** El método POST se utiliza más a menudo para “crear” recursos nuevos. Es decir, cuando el método se ejecuta sin errores, se crea una nueva representación del recurso enviado se le asigna un “id” o identificador, y se puede devolver el nuevo recurso o un simple mensaje como respuesta además de un código de respuesta HTTP 200 (ok). POST no es seguro ni idempotente. Por lo tanto, se recomienda para solicitudes de recursos no idempotentes. Hacer dos solicitudes POST idénticas probablemente resulte en dos recursos que contengan la misma información.
- **PUT:** La mayoría de las veces el método PUT se utiliza para “actualizar” todo un recurso; enviando a un URI de recursos conocido la representación recién actualizada del recurso original. Sin embargo, PUT también se puede utilizar para crear un recurso en el caso en que el cliente elija el “id” del recurso en lugar de hacerlo el servidor. En otras palabras, si el PUT es a un URI que contiene el valor de un “id” de recurso inexistente este creará una nueva entidad. Muchos sienten que esto es intrincado y confuso. En consecuencia, este método de creación debe usarse con moderación, si es que lo hace.
- **PATCH:** El método PATCH se usa para “modificar” recursos. La solicitud PATCH solo necesita contener los cambios en el recurso, no el recurso completo. Esto se asemeja al método PUT, pero el cuerpo de la información contiene un conjunto de instrucciones que describen cómo debe modificarse un recurso que actualmente reside en el servidor para producir una nueva versión.

- **DELETE:** El método DELETE es bastante fácil de entender, se usa para "eliminar" un recurso identificado por un URI. En la eliminación exitosa, regresa el estado HTTP 200 (ok) junto con un cuerpo de respuesta, tal vez la representación del elemento eliminado o el estado HTTP 204 (sin contenido) sin cuerpo de respuesta. Las operaciones DELETE son idempotentes, es decir, si “elimina” un recurso, se elimina, llamar repetidamente a DELETE sobre ese recurso termina igual: el recurso ha sido eliminado. Sin embargo, DELETE también puede devolver un 404 (no encontrado) si se llama por segunda o más veces, ya que ya se eliminó y, por lo tanto, ya no se puede encontrar.

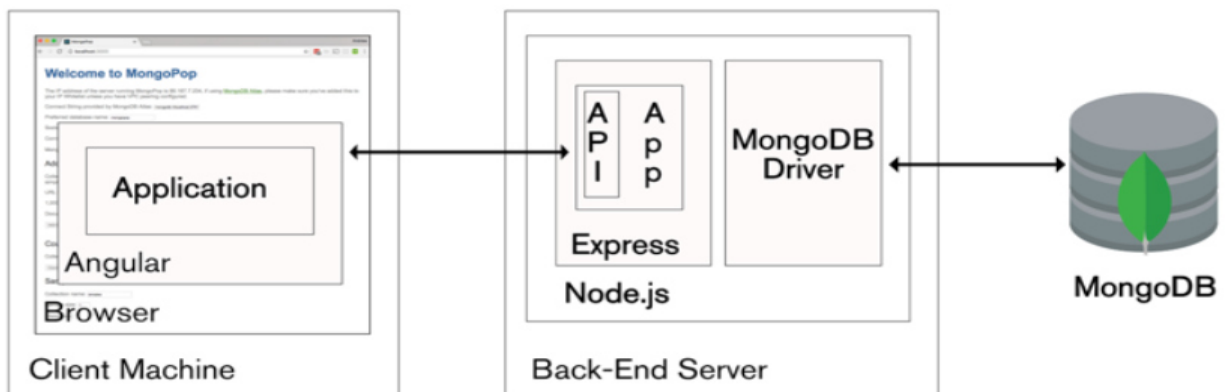
FRAMEWORK MEANJS

El concepto surgió de Amos Haviv, un programador web israelí de la compañía israelí Linnovate, quien notó una tendencia de sus clientes de integrar sistemas implementados en JavaScript, MongoDB, Node.js y ExpressJS en la parte del servidor o backend y Angular en la parte del cliente o frontend. Sin embargo, fue Valeri Karpov quien había propuesto el acrónimo “MEAN” (MongoDB, ExpressJS, AngularJS y Node.js) en un blog escrito por ella (MEANJS, 2018).

Amos Haviv presentó una versión de MEAN y junto con Linnovate desarrollaron lo que hoy se conoce como “MeanIO”. Por diferencias en el rumbo que estaba tomando el proyecto, Amos Haviv decide abandonar su colaboración con Linnovate y decide crear “MeanJS” como alternativa a “MeanIO” (MEANJS, 2018).

Mientras que “MeanIO” sigue siendo mantenido y desarrollado por Linnovate, “MeanJS” es mantenido y desarrollado por una comunidad de expertos y aficionados en el tema por medio de internet.

Figura 7
Framework MEANJS



Fuente: (MongoDB, 2017)

MONGODB

MongoDB es una base de datos de documentos de código abierto (open source) y está diseñada teniendo en cuenta tanto la escalabilidad como la agilidad del desarrollador. MongoDB cierra la brecha entre los almacenes de clave valor, que son rápidos y escalables, y las bases de datos relacionales, que tienen una gran funcionalidad. En lugar de almacenar datos en filas y columnas como lo haría con una base de datos relacional, MongoDB almacena documentos JSON en colecciones con esquemas dinámicos (MongoDB, 2017).

Figura 8
Ejemplo de un documento JSON

```
{
  _id: ObjectId("95497f06c53c11e8a355529269fb1459"),
  name: "Joe Bookreader",
  address: {
    street: "123 Fake Street",
    city: "Faketon",
    state: "MA",
    zip: "12345"
  }
}
```

Fuente: (MongoDB, 2018)

EXPRESSJS

Es un framework de aplicaciones web que ejecuta la aplicación backend (JavaScript). Express es una dependencia de Node.js y se ejecuta como un módulo (ExpressJS, 2018).

Puede manejar el enrutamiento de las solicitudes realizadas por el cliente (frontend) hacia las partes internas de la aplicación o aplicaciones externas si así se requiere.

Además, puede generar el HTML final para que el navegador del cliente lo muestre o actuar simplemente como un API REST, lo que permite a la aplicación frontend acceder a los recursos que necesite (proceso de datos, acceso a la base de datos, etc.).

Figura 9

Ejemplo de API REST con ExpressJS

```
app.get('/', function (req, res) {  
  res.send('Hello World!');  
});  
  
app.post('/', function (req, res) {  
  res.send('Got a POST request');  
});  
  
app.put('/user', function (req, res) {  
  res.send('Got a PUT request at /user');  
});  
  
app.delete('/user', function (req, res) {  
  res.send('Got a DELETE request at /user');  
});
```

Fuente: (ExpressJS, 2018)

ANGULARJS

Creado y mantenido por Google, ejecuta código JavaScript dentro de los navegadores web del cliente para implementar una interfaz de usuario (UI)

dinámica, a diferencia de los formularios web estáticos que al realizar una acción se tiene que esperar (AngularJS, 2018).

Angular (nombre corto de AngularJS), implementa la aplicación Frontend como un conjunto de componentes, los cuales consisten en código JavaScript y una plantilla HTML que incluye elementos para mostrar los datos o ejecutar funciones descritas en el JavaScript.

Figura 10

Ejemplo de Implementación con AngularJS

```
1. <!doctype html>
2. <html ng-app>
3.   <head>
4.     <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.4/
angular.min.js"></script>
5.   </head>
6.   <body>
7.     <div>
8.       <label>Name:</label>
9.       <input type="text" ng-model="yourName" placeholder="Enter a name here">
10.     <hr>
11.     <h1>Hello {{yourName}}!</h1>
12.   </div>
13. </body>
14. </html>
```

Fuente: (AngularJS, 2018)

NODE.JS

Node.js es un entorno de ejecución de JavaScript orientado a eventos asíncronos, y se basa en el motor de JavaScript V8 de Google que se utiliza en los navegadores Chrome. También incluye una serie de módulos que proporcionan funciones esenciales para implementar aplicaciones web, incluidos protocolos de red como HTTP. Además, se pueden instalar módulos de terceros, incluido el controlador MongoDB, utilizando la herramienta “npm” (Node.js, 2018).

Al ser asíncrono la aplicación ejecuta una solicitud y continúa trabajando en otras tareas sin estancarse a la espera de una respuesta a la solicitud. Al completarse la solicitud, se informa de a través de una devolución de llamada.

Esto permite realizar un gran número de operaciones en paralelo, lo que es esencial al escalar aplicaciones. MongoDB también realiza operaciones de forma asíncrona, por lo que funciona muy bien con este tipo de aplicaciones (MongoDB, 2018).

Figura 11

Ejemplo de un Servidor con Node.js

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Fuente: (Node.js, 2018)

3.2.5 IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS

INICIO

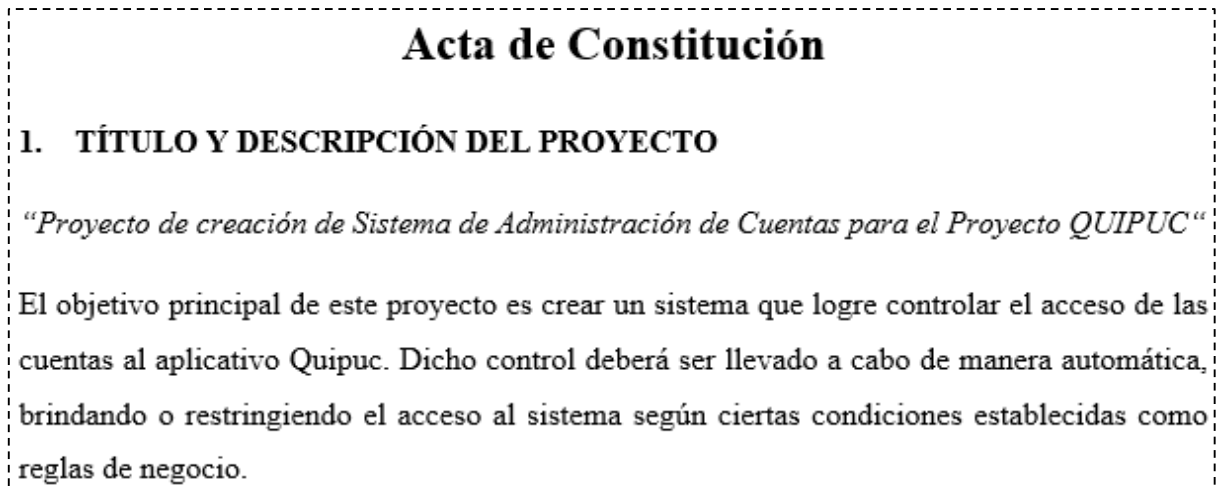
De acuerdo al cronograma general de la empresa, se plantea realizar una reunión entre el Director General y el Jefe de Proyectos Web para establecer los lineamientos del proyecto a realizar.

La primera reunión se llevó a cabo el 27 de diciembre donde participaron el Director General, el Jefe de Proyectos Web y el Analista Funcional, el motivo se debió a que el aplicativo Quipuc ya había lanzado su versión “beta” y ya era necesario comenzar a cobrar por su uso. Se planteó la idea de desarrollar un proyecto en paralelo que cumpla con dicha necesidad.

En las reuniones posteriores, que fueron realizados en días continuados, se elaboró el Acta de Constitución del Proyecto y con la lista de requerimientos se elaboró las Historias de Usuario (ver Anexo 3).

Figura 12

Acta de Constitución del Proyecto



Fuente: (Proyecto de Administración de Cuentas, 2018)

El rol de Product Owner fue tomado por el Director General, a su vez el asignó al mismo Jefe de Proyectos Web el rol de Scrum Master, para que llevara a cabo el proyecto de la mejor manera dado su buena experiencia en estos tipos de proyectos. Se planteó una última reunión para definir el equipo y así dar inicio a la ejecución del proyecto.

PLANIFICACIÓN Y REQUERIMIENTOS

Como primera tarea de esta etapa se plantea una reunión para realizar la priorización de lo recopilado en el Product Backlog (ver Tabla 2) comparándolo siempre con las Historias de Usuario por si existiese alguna inconsistencia, por lo que se optó por asignarle un código según la complejidad, similitud y diferencias que pudiesen tener. El código a utilizar es un nuevo formato y tiene la siguiente forma:

PBI - OXYZ

Donde PBI significa “Product Backlog Item” y XY representa al módulo y Z la tarea a realizar. A la letra “O” se le asignará un valor en caso esta tarea requiera la coordinación con otro proyecto, en este caso Quipuc.

Tabla 2
Product Backlog

Código	Funcionalidad	Descripción
PBI-0010	Loguear a la aplicación.	
PBI-0020	Crear un demonio para que se ejecute mensualmente.	Crear un demonio para que se ejecute en un determinado día del mes (por defecto el día 1 a 00:00 horas de cada mes).
PBI-0021	Calcular la deuda, de cada negocio, según usuarios activos.	Calcular la deuda, de cada negocio, según usuarios activos y enviar un correo notificando la información de la deuda.
PBI-0022	Enviar una notificación con la información de la deuda.	
PBI-0023	Generar un excel con la lista de negocios junto con su respectiva deuda.	Generar un excel con la lista de negocios con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta, Estado de Pago; y enviarlo por email.
PBI-0024	Enviar por email el excel generado con la información de los negocios.	
PBI-0025	“Notificar” a los negocios.	Cambiar de estado a “notificado” a todos los negocios que tengan el estado “activo”.
PBI-0030	Cargar un excel con información de los negocios.	Crear una tarea para la lectura de un excel con un formato definido (similar al formato mencionado anteriormente). El excel será guardado para un posterior uso.
PBI-0040	Crear un demonio para que se ejecute diariamente.	Crear un demonio para que se ejecute en una determinada hora de cada día (por defecto a las 00:00 horas).
PBI-0041	“Activar” a los negocios	Cambiar al estado “activo” a los negocios si cierta columna (del excel cargado anteriormente) tiene un valor definido para su activación.
PBI-0050	Crear un demonio para que se ejecute mensualmente.	Crear un demonio para que se ejecute en un determinado día del mes (por defecto el día 17 a 00:00 horas del mes en transcurso).

PBI-0051	“Bloquear” a los negocios.	Cambiar de estado a “bloqueado” a todos los negocios que tengan el estado 'notificado'.
PBI-0060	Exportar estados de cuenta.	Generar un excel con la lista de negocios con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta, Estado de Pago; y enviarlo por email.
PBI-0070	Listar Negocios.	Listar en la vista una tabla con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta y Estado de Pago, Nombre del Distribuidor, Código de Distribuidor (Editable).
PBI-0080	Buscar Negocio.	Permitir buscar un negocio en específico mediante su nombre o su email principal.
PBI-0090	Ingresar al Quipuc del negocio.	Ingresar automáticamente a la aplicación web de Quipuc correctamente autenticado. Los permisos deben ser de tipo “Dueño”
PBI-0100	Editar Negocio.	Mostrar el detalle del Negocio como: Nombre del Negocio, Email del Dueño, Estado de Pago, Nombre del Distribuidor, Estado de Cuenta y Código de Distribuidor; donde estos 2 últimos serán los únicos editables.
PBI-1000	Ingresar como “Super Admin”.	Para acceder a este modo, debe ser redirigido desde Quipuc Admin.
PBI-1010	Añadir un nuevo campo de referido al registrar un negocio.	Añadir un nuevo campo llamado “Código de Referido” que hará referencia al distribuidor. (Paso 2 del Registro).
PBI-1020	Realizar operaciones en Quipuc como dueño.	Manipular la información del negocio seleccionado en “Quipuc Admin” como rol “Dueño”.

Fuente: Elaboración Propia.

Luego de este análisis se optó por priorizar y agrupar las tareas por código y elaborar, según su prioridad, el Sprint Backlog. Según esto, el desarrollo del proyecto se realiza en 5 iteraciones o Sprints con una duración de 1 semana para cada Sprint (ver Tabla 3).

Tabla 3
Sprint Backlog Resumido

Sprint	Inicio	Fin	Códigos
Sprint 1	09/01/17	13/01/17	PBI-0010 - PBI-0020 - PBI-0021 - PBI-0022
Sprint 2	16/01/17	20/01/17	PBI-0023 - PBI-0024 - PBI-0025 - PBI-0030 - PBI-0040 - PBI-0041
Sprint 3	23/01/17	27/01/17	PBI-0050 - PBI-0051 - PBI-0060 - PBI-0070 - PBI-0080
Sprint 4	30/01/17	02/02/17	PBI-0090 - PBI-0100 - PBI-1000
Sprint 5	04/02/17	08/02/17	PBI-1010 - PBI-1020

Fuente: Elaboración Propia.

Una vez elaborado el Sprint Backlog, se procedió a elaborar el cronograma del proyecto (ver Anexo 6) en base a las estimaciones en las tareas priorizadas en el Sprint Backlog (ver Anexo 5).

El Scrum diario se estableció para que diera lugar a las 10:00 am en la sala de reuniones, las reuniones con el Product Owner se realizarán previa coordinación pero se propuso que sea al iniciar la semana, aproximadamente a las 11:00 am, debido a posibles reuniones de mayor prioridad que surjan para ese día.

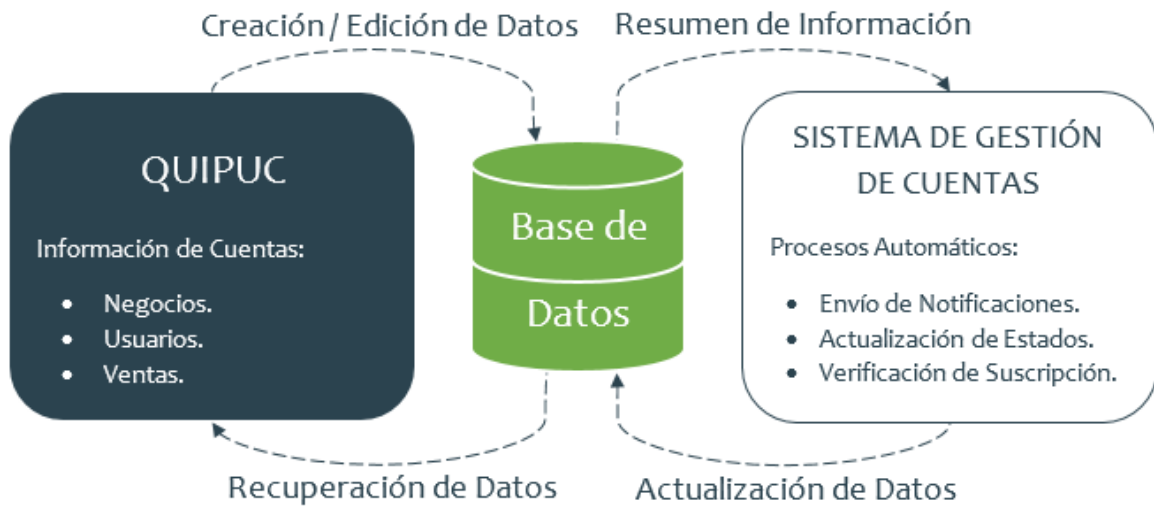
ANÁLISIS Y DISEÑO

En esta etapa se procede a analizar el comportamiento y la interacción que va tener el nuevo sistema con Quipuc. Debido a que ambos sistemas van a compartir la misma Base de Datos, la información que requiera un sistema del otro va a estar disponible inmediatamente (ver Figura 13).

Por ende, ya no fue necesario realizar muchas modificaciones al módulo de autenticación, ya que solo sería necesario validar el estado de la cuenta luego de validar las credenciales del usuario (ver Figura 14).

Figura 13

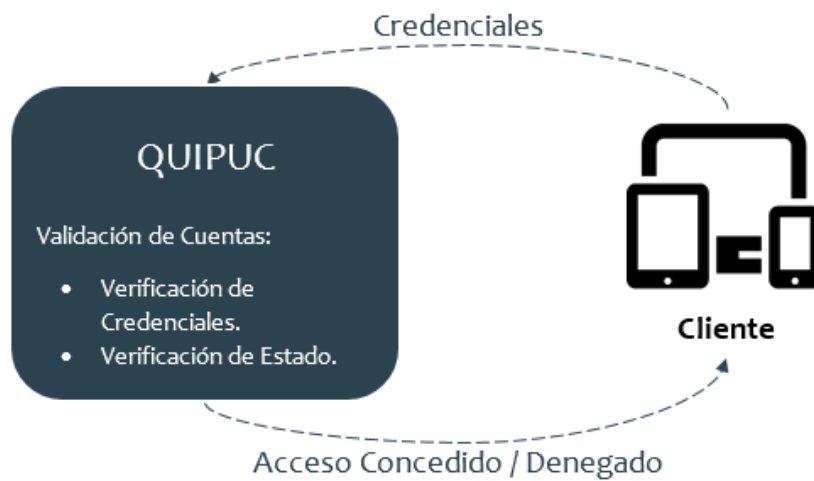
Interacción Quipuc, Base de Datos y el Nuevo Sistema



Fuente: Elaboración Propia.

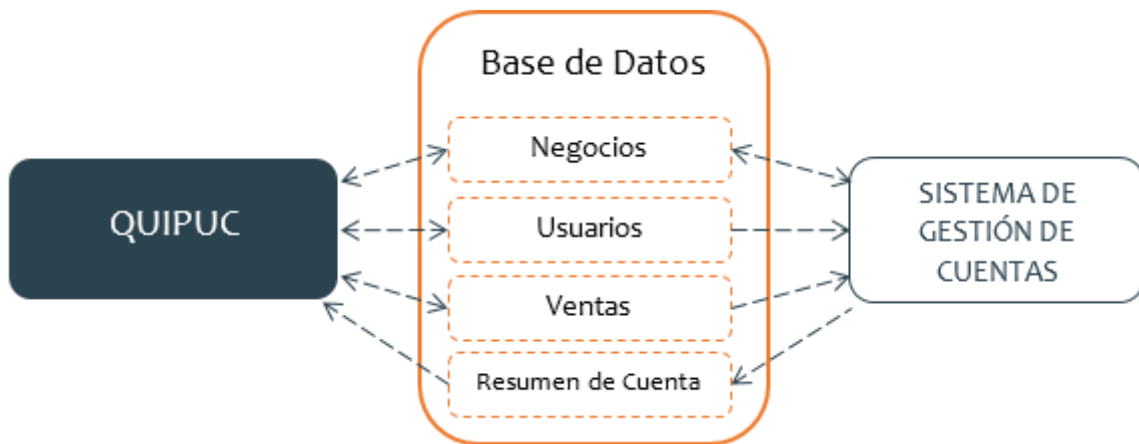
Figura 14

Inicio de Sesión en Quipuc



Fuente: Elaboración Propia.

Figura 15
Acceso de Datos



Fuente: Elaboración Propia.

Con esta información se procede a mapear los recursos que serán utilizados por Quipuc y el Sistema de Administración de Cuentas. Los datos de los negocios están disponibles y pueden ser utilizados y actualizados por ambos sistemas, pero los recursos de usuarios y ventas solo podrán ser actualizados por Quipuc. Por otra parte, el resumen de la cuenta será generada periódicamente por el Sistema de Administración de Cuentas y estará disponible a modo de consulta para Quipuc.

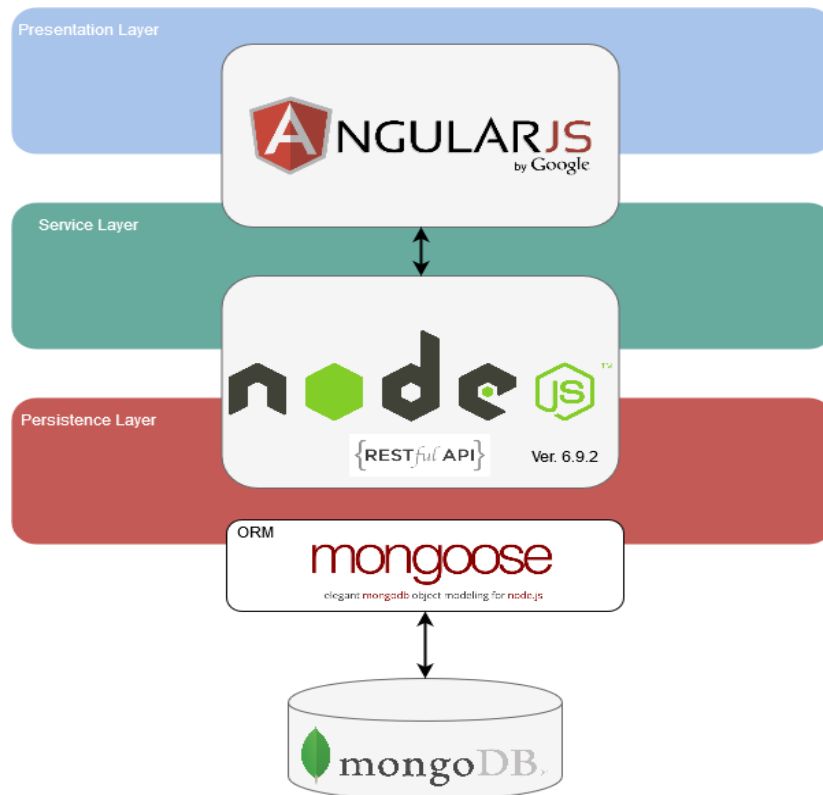
Luego se procedió a elaborar la arquitectura del proyecto, orientado solamente para web, era opcional que tenga un comportamiento responsive pero no tenía prioridad ya que el core del proyecto se basa en la parte del servidor. La arquitectura final consta de 3 capas principales, 2 orientadas a la parte del cliente o Frontend y 1 en la parte del servidor o Backend (ver Figura 16).

Por la parte del servidor, se maneja la comunicación del servidor con la base de datos mediante la dependencia “mongoose” y la dependencia “express” para levantar el servicio RESTful, ambas dependencias cuentan con una buena aceptación en los usuarios de Node.js (ver Figura 17).

La arquitectura del cliente son dos capas: Las vistas y los servicios que se comunican con el servidor (ver Figura 18). La tecnología a utilizar en esta parte del proyecto será AngularJs.

Figura 16

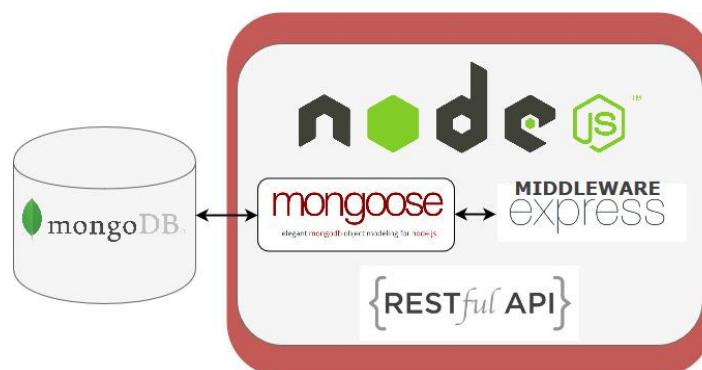
Arquitectura del Proyecto



Fuente: (Proyecto de Administración de Cuentas, 2018)

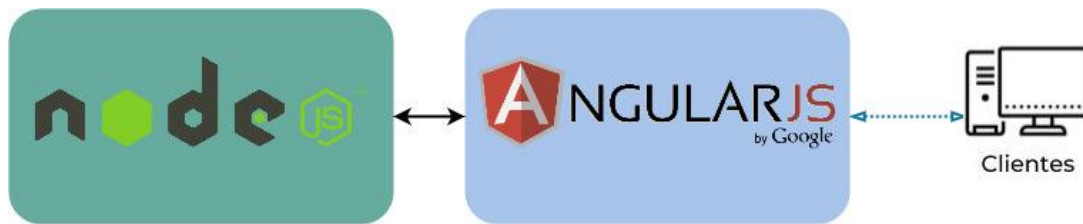
Figura 17

Arquitectura del Servidor



Fuente: Elaboración Propia.

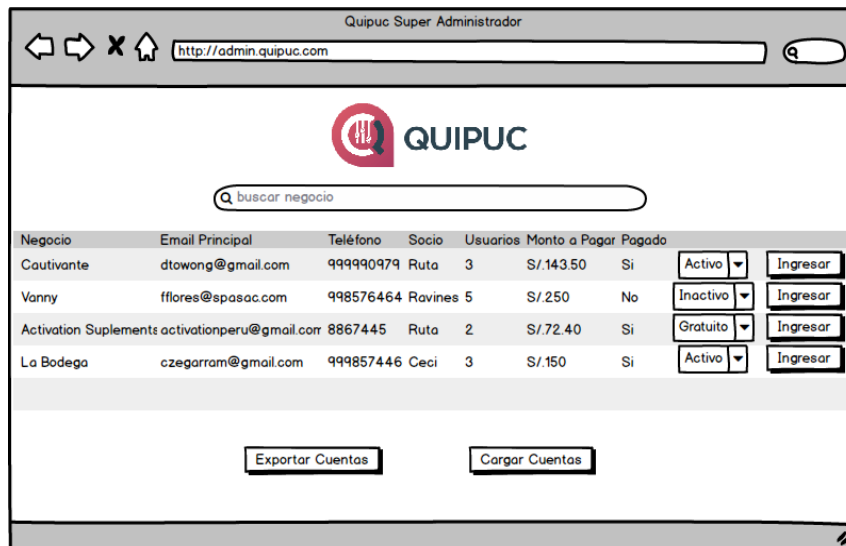
Figura 18
Arquitectura del Cliente



Fuente: Elaboración Propia.

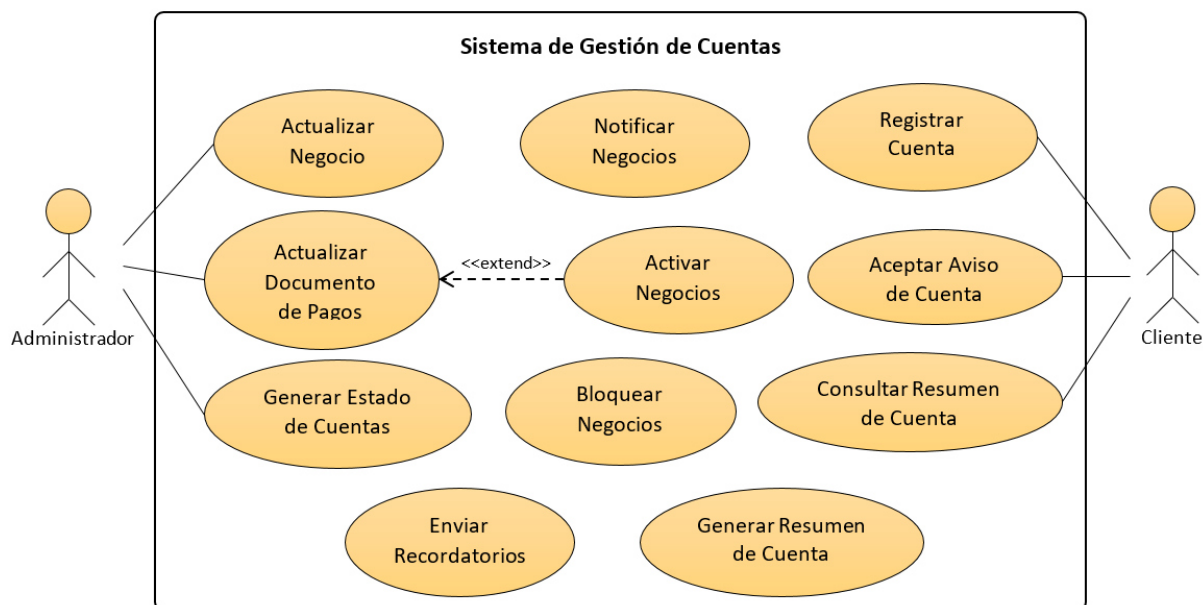
El Frontend o cliente de la aplicación manejan 2 vistas: La vista de autenticación o “login” y la vista general, de ésta última fue la única a la que se le elaboró un prototipo (ver Figura 19), este contiene información básica de los negocios.

Figura 19
Prototipo - Vista General



Fuente: (Proyecto de Administración de Cuentas, 2018)

Figura 20
Diagrama de Casos de Uso



Fuente: Elaboración Propia.

Se procedió a elaborar los casos de uso y el diagrama de casos de uso (ver Figura 20) para observar mejor la relación que tienen los casos de uso y el autor principal, el “administrador”, quien podrá realizar o ejecutar 3 casos de uso en el sistema y son:

- **Actualizar Negocio:** Edición simple de los datos del negocio como el estado, nombre de asociado, etc.
- **Generar Estado de Cuentas:** Importación en un archivo de los negocios con sus respectivas cuentas pendientes o no de pago, cantidad de usuarios y demás datos.
- **Actualizar Documento de Pagos:** Subida de un archivo de cálculo que contiene la información de los negocios y sus respectivos pagos, según un formato con el que trabaja la empresa. Existe la opción de ejecutar el caso de uso de “Activar Negocios”, pero ya es responsabilidad del “administrador” si decide ejecutar dicha opción.

El siguiente actor es el “cliente”, quien podrá realizar o ejecutar 3 casos de uso en el sistema y son:

- **Registrar Cuenta:** Registro de los datos básicos del negocio, .del usuario, configuraciones iniciales, etc.
- **Aceptar Aviso de Cuenta:** Verificación de cuenta al culminar el número de días de prueba asignados a una cuenta.
- **Consultar Resumen de Cuenta:** Acceso al historial generado en el sistema, sin embargo, también se genera automáticamente si el usuario consulta datos de la semana actual.

Sin embargo, existen 5 casos de uso que representan las tareas que realiza el mismo sistema como procesos automáticos:

- **Notificar Negocios:** Calcula el costo de la deuda del negocio, notifica al dueño del negocio mediante un mensaje de correo electrónico y cambia de estado del negocio a “notificado”.
- **Activar Negocios:** Realiza la lectura del archivo de cálculo subido anteriormente, verifica su formato, actualiza los pagos de los negocios y cambia el estado de los negocios a “activo”.
- **Bloquear Negocios:** Cambia el estado de los negocios a “bloqueado” a quienes que no hayan llegaron a pagar por completo su deuda.
- **Enviar Recordatorios:** Envío de información como tutoriales o datos del negocio para mejorar la interacción del usuario con el sistema Quipuc, además de recordarle la cantidad de días de prueba que le restan a su cuenta.
- **Generar Resumen de Cuenta:** Recopilación semanal de información del negocio como ventas (montos y cantidad de ventas), cantidad de usuarios, ingresos, etc.

Describimos a continuación los Casos de Uso mostrados en el D.C.U.

Tabla 4
Caso de Uso: Actualizar Negocio

Nombre	CU – 01. Actualizar Negocio
Precondiciones	<ul style="list-style-type: none"> - El Usuario “Administrador” debe estar logueado en el sistema. - El negocio a editar debe existir en el repositorio de datos.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Usuario “Administrador” elige un Negocio de la lista de negocios. 2. El Sistema carga la información del negocio del repositorio de datos. 3. El Usuario modifica los campos disponibles de edición. 4. El Usuario elige la opción “Guardar”. 5. El Sistema modifica los campos editados por el usuario y los almacena en el repositorio de datos. 6. El caso de uso culmina al recibir un mensaje de éxito por parte del sistema.
Error / Secuencia Alternativa	<ol style="list-style-type: none"> 4.a. Si el Usuario elige la opción “Cancelar” entonces el caso de uso culmina sin emitir ningún mensaje. 5.e. Si ocurre un error al realizar el proceso de guardado, el sistema muestra un mensaje de error y retorna al paso 4.
Resultado	<ol style="list-style-type: none"> 6. El Sistema muestra la pantalla principal. 4.a. El Sistema muestra la pantalla principal.

Fuente: Elaboración Propia.

Tabla 5
Caso de Uso: Actualizar Documento de Pagos

Nombre	CU – 02. Actualizar Documento de Pagos
Precondiciones	<ul style="list-style-type: none"> - El Usuario “Administrador” debe estar logueado en el sistema.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema muestra una lista de archivos del sistema de archivos que tenga el usuario. 2. El Usuario “Administrador” elige un archivo con extensión “xls”. 3. El Usuario elige la opción “Cargar”. 4. El Sistema verifica la información y los campos contenidos en el archivo elegido por el usuario. 5. El Sistema guarda el archivo elegido por el usuario en su repositorio de archivos. 6. El Usuario elige la opción “Cerrar”. 7. El caso de uso culmina al recibir un mensaje de éxito por parte del sistema.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 3.a. Si el Usuario elige la opción “Cancelar” entonces el caso de uso culmina sin emitir ningún mensaje. 4.e. Si ocurre un error al realizar el proceso de validación, el sistema muestra un mensaje de error y retorna al paso 2. 5.e. Si ocurre un error al realizar el proceso de guardado, el sistema muestra un mensaje de error y retorna al paso 2. 6.a. Si el Usuario elige la opción “Activar Negocios” se procede a ejecutar el Caso de Uso: Activar Negocios.
Resultado	<ol style="list-style-type: none"> 7. El Sistema muestra la pantalla principal. 6.a. El Sistema muestra la pantalla principal. 3.a. El Sistema muestra la pantalla principal.

Fuente: Elaboración Propia.

Tabla 6
Caso de Uso: Generar Estado de Cuentas

Nombre	CU – 03. Generar Estado de Cuentas
Precondiciones	- El Usuario “Administrador” debe estar logueado en el sistema.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Usuario “Administrador” elige la opción “Generar Estado de Cuentas” 2. El Sistema genera un archivo de cálculo, según la lista de negocios disponibles, con información relevante de los negocios junto con sus respectivos Estados de Cuenta. 3. El Sistema envía el archivo de cálculo generado a la cuenta de correo electrónico del usuario. 4. El caso de uso culmina al recibir un mensaje de éxito por parte del sistema.
Error / Secuencia Alterna	<p>2.e. Si ocurre un error al realizar el proceso de generación de archivo, el sistema muestra un mensaje de error y retorna al paso 1.</p> <p>3.e. Si ocurre un error al realizar el proceso de envío de archivo, el sistema muestra un mensaje de error y retorna al paso 1.</p>
Resultado	4. El Sistema muestra la pantalla principal.

Fuente: Elaboración Propia.

Tabla 7
Caso de Uso: Notificar Negocios

Nombre	CU – 04. Notificar Negocios
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema selecciona a todos los negocios con estado “activo”. 2. El Sistema calcula la cantidad de cuentas de usuarios activas que tiene cada negocio. 3. El Sistema calcula el costo de suscripción según la cantidad de cuentas que tenga cada negocio y fecha de creación de dichas cuentas. 4. El Sistema actualiza el estado a “notificado” y registra el costo calculado en el Estado de Cuenta (pagos pendientes) de todos los negocios procesados. 5. El Sistema envía un correo electrónico a los negocios procesados con su respectiva información de estado de cuenta. 6. El Sistema genera un archivo de cálculo, según la lista de negocios procesados, con información relevante de los negocios junto con sus respectivos Estados de Cuenta. 7. El Sistema envía el archivo de cálculo generado a una cuenta de correo electrónico en específico. 8. El caso de uso culmina al registrar el proceso realizado en el log de procesos automáticos.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 1.a. Si la cantidad de negocios con estado “activo” es igual a 0, el sistema genera un mensaje y se dirige al paso 8. 3.e. Si ocurre un error al realizar el proceso de cálculo de deuda, el sistema genera un mensaje de error, retorna al paso 3 y procesa al siguiente negocio. 4.e. Si ocurre un error al realizar el proceso de actualización, el sistema genera un mensaje de error y se dirige al paso 8.

	<p>5.e. Si ocurre un error al realizar el proceso de envío de correo electrónico, el sistema muestra un mensaje de error y se dirige al paso 8.</p> <p>6.e. Si ocurre un error al realizar el proceso de generación del archivo de cálculo, el sistema muestra un mensaje de error y se dirige al paso 8.</p> <p>7.e. Si ocurre un error al realizar el proceso de envío de correo electrónico, el sistema muestra un mensaje de error y se dirige al paso 8.</p>
--	---

Fuente: Elaboración Propia.

Tabla 8
Caso de Uso: Activar Negocios

Nombre	CU – 05. Activar Negocios
Precondiciones	- El Caso de Uso: CU – 02. Actualizar Documento de Pagos, debe haber culminado exitosamente.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema buscar el archivo subido de su repositorio de archivos. 2. El Sistema realiza la lectura del archivo de cálculo buscando los negocios listados. 3. El Sistema actualiza el costo de la deuda y el estado a “activo” de los negocios encontrados en el archivo de cálculo. 4. El caso de uso culmina al registrar el proceso realizado en el log de procesos automáticos.
Error / Secuencia Alternativa	<p>1.a. Si el archivo no existe, el sistema genera un mensaje y se dirige al paso 4.</p> <p>2.e. Si ocurre un error al realizar el proceso de lectura del archivo, el sistema genera un mensaje de error y se dirige al paso 4.</p>

	3.e. Si ocurre un error al realizar el proceso de actualización, el sistema genera un mensaje de error y se dirige al paso 4.
--	---

Fuente: Elaboración Propia.

Tabla 9

Caso de Uso: Bloquear Negocios

Nombre	CU – 06. Bloquear Negocios
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema selecciona a todos los negocios con estado “notificado”. 2. El Sistema actualiza el estado a “bloqueado” a todos los negocios encontrados. 3. El caso de uso culmina al registrar el proceso realizado en el log de procesos automáticos.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 1.a. Si la cantidad de negocios con estado “notificado” es igual a 0, el sistema genera un mensaje y se dirige al paso 3. 2.e. Si ocurre un error al realizar el proceso de actualización, el sistema genera un mensaje de error y se dirige al paso 3.

Fuente: Elaboración Propia.

Tabla 10
Caso de Uso: Enviar Recordatorios

Nombre	CU – 07. Enviar Recordatorios
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema selecciona a todos los negocios que no tengan asignado un estado o tengan el estado “ninguno”. 2. El Sistema calcula la cantidad de días de prueba que tenga la cuenta. 3. El Sistema genera el mensaje a enviar según la fecha de creación del negocio y la configuración ingresada. 4. El sistema envía el mensaje por correo electrónico a los negocios procesados. 5. El caso de uso culmina al registrar el proceso realizado en el log de procesos automáticos.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 1.a. Si la cantidad de negocios que no tienen estado o con estado “ninguno” es igual a 0, el sistema genera un mensaje y se dirige al paso 5. 3.e. Si ocurre un error al generar el mensaje personalizado, el sistema retorna al paso 3 y procesa al siguiente negocio. 4.e. Si ocurre un error al realizar el envío de correos electrónicos, el sistema genera un mensaje de error y se dirige al paso 5.

Fuente: Elaboración Propia.

Tabla 11

Caso de Uso: Generar Resumen de Cuenta

Nombre	CU – 08. Generar Resumen de Cuenta
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema selecciona a todos los negocios con estado diferente a “bloqueado”. 2. El Sistema selecciona las ventas realizadas en la presente semana de los negocios encontrados. 3. El Sistema genera el resumen de cuenta según las ventas realizadas, monto de las ventas, usuarios, etc. 4. El Sistema guarda en el historial el resumen de cuenta generado. 5. El caso de uso culmina al registrar el proceso realizado en el log de procesos automáticos.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 1.a. Si la cantidad de negocios con estado diferente a “bloqueado” es igual a 0, el sistema genera un mensaje y se dirige al paso 5. 3.e. Si ocurre un error al generar el resumen de cuenta, el sistema genera un mensaje de error, retorna al paso 3 y procesa al siguiente negocio. 4.e. Si ocurre un error al realizar el proceso de guardado, el sistema genera un mensaje de error y se dirige al paso 5.

Fuente: Elaboración Propia.

Tabla 12
Caso de Uso: Registrar Cuenta

Nombre	CU – 09. Registrar Cuenta
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema muestra el formulario de registro. 2. El Usuario “Cliente” ingresa los datos básicos de su negocio (Nombre Comercial, Razón Social, RUC, correo electrónico, contraseña, etc.). 3. El Usuario “Cliente” selecciona la opción “Aceptar Términos y Condiciones”. 4. El Usuario “Cliente” selecciona la opción “Registrar”. 5. El Sistema guarda los datos ingresados generando una nueva cuenta. 6. El caso de uso culmina al recibir un mensaje de éxito por parte del sistema.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 3.a. En caso de no seleccionar la opción “Aceptar Términos y Condiciones”, el caso de uso culmina sin emitir ningún mensaje. 4.a. Si el Usuario “Cliente” selecciona la opción “Cancelar”, el caso de uso culmina sin emitir ningún mensaje. 5.e. Si ocurre un error al realizar el proceso de guardado, el sistema genera un mensaje de error y se dirige al paso 4.
Resultado	<ol style="list-style-type: none"> 6. El Sistema muestra la pantalla inicial. 3.a. El Sistema muestra la pantalla inicial. 4.a. El Sistema muestra la pantalla inicial.

Fuente: Elaboración Propia.

Tabla 13

Caso de Uso: Aceptar Aviso de Cuenta

Nombre	CU – 10. Aceptar Aviso de Cuenta
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema muestra la vista de inicio de sesión. 2. El Usuario “Cliente” ingresa sus credenciales. 3. El Sistema verifica el estado del negocio de la cuenta y si ya culminaron sus días de prueba. 4. El Sistema muestra el Aviso de Activación de Cuenta. 5. El Usuario “Cliente” selecciona la opción “Activar Cuenta”. 6. El Sistema cambia el estado del negocio a “activo”. 7. El caso de uso culmina al recibir un mensaje de éxito por parte del sistema.
Error / Secuencia Alterna	<ol style="list-style-type: none"> 2.e. Si ocurre un error al realizar el proceso de validación, el sistema muestra un mensaje de error y retorna al paso 2. 3.a. Si la cuenta aún cuenta con días de prueba, el caso de uso culmina sin emitir ningún mensaje. 5.a. Si el Usuario “Cliente” selecciona la opción “Cerrar”, el caso de uso culmina sin emitir ningún mensaje. 5.e. Si ocurre un error al realizar el proceso de activación, el sistema genera un mensaje de error y se dirige al paso 5.
Resultado	<ol style="list-style-type: none"> 6. El Sistema muestra la pantalla principal. 3.a. El Sistema muestra la pantalla principal. 5.a. El Sistema muestra la pantalla inicial.

Fuente: Elaboración Propia.

Tabla 14

Caso de Uso: Consultar Resumen de Cuenta

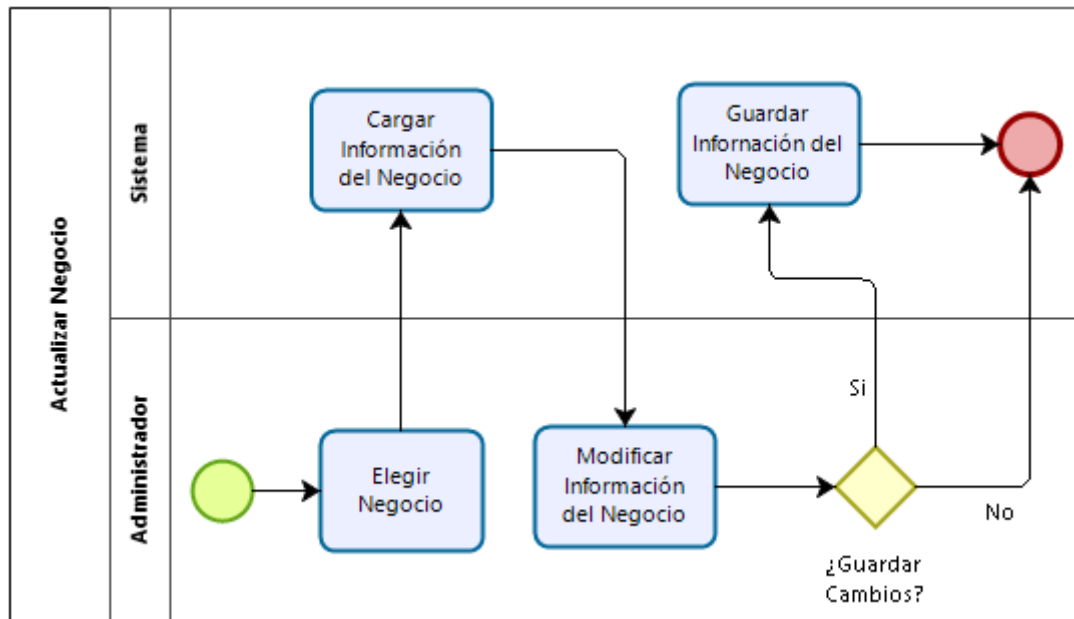
Nombre	CU – 11. Consultar Resumen de Cuenta
Precondiciones	Ninguna.
Secuencia Principal	<ol style="list-style-type: none"> 1. El Sistema muestra una lista de intervalos de fechas que corresponden a semanas y la semana actual. 2. El Usuario “Cliente” selecciona un ítem de la lista. 3. El Sistema muestra la información guardada correspondiente al ítem seleccionado (semana). 4. El caso de uso culmina cuando el Usuario “Cliente” selecciona la opción “Cerrar”.
Error / Secuencia Alternativa	2.a. Si el Usuario “Cliente” selecciona la opción “Semana Actual”, el sistema genera el resumen de cuenta con datos de la presente semana (ventas, monto de ventas, cantidad de usuarios, etc.).
Resultado	4. El Sistema muestra la pantalla principal.

Fuente: Elaboración Propia.

En base a la descripción de cada Caso de Uso, se elaboraron los diagramas de actividades que corresponden a cada proceso que se espera implementar en el nuevo sistema.

Figura 21

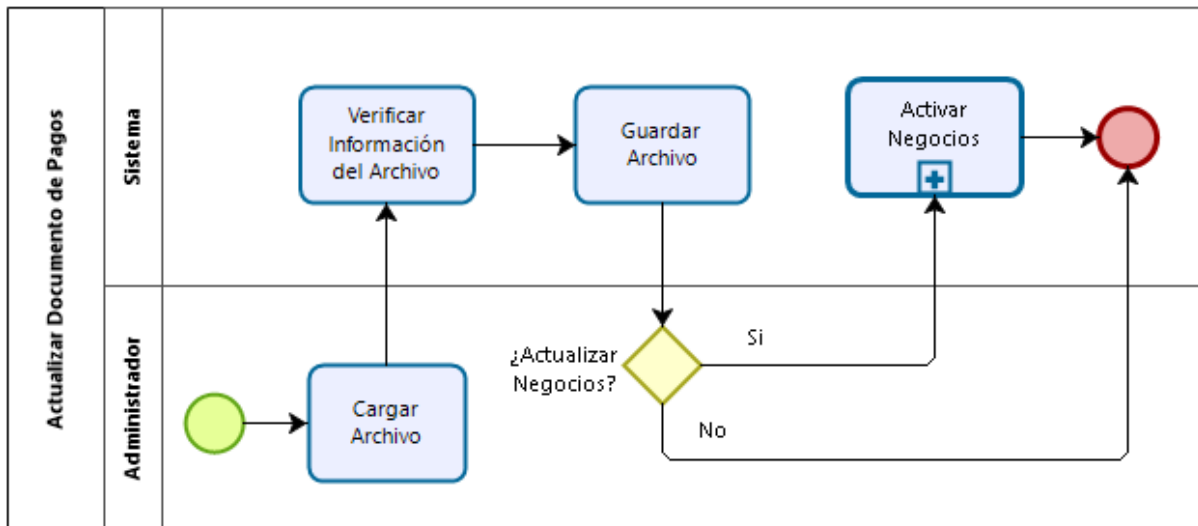
Diagrama de Actividades: Actualizar Negocio



Fuente: Elaboración Propia.

Figura 22

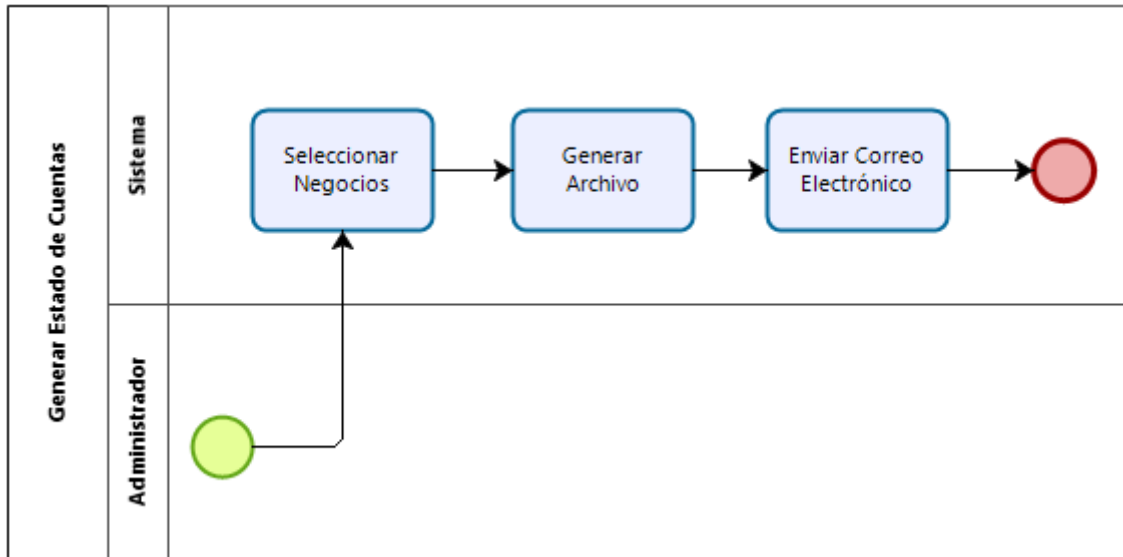
Diagrama de Actividades: Actualizar Documento de Pagos



Fuente: Elaboración Propia.

Figura 23

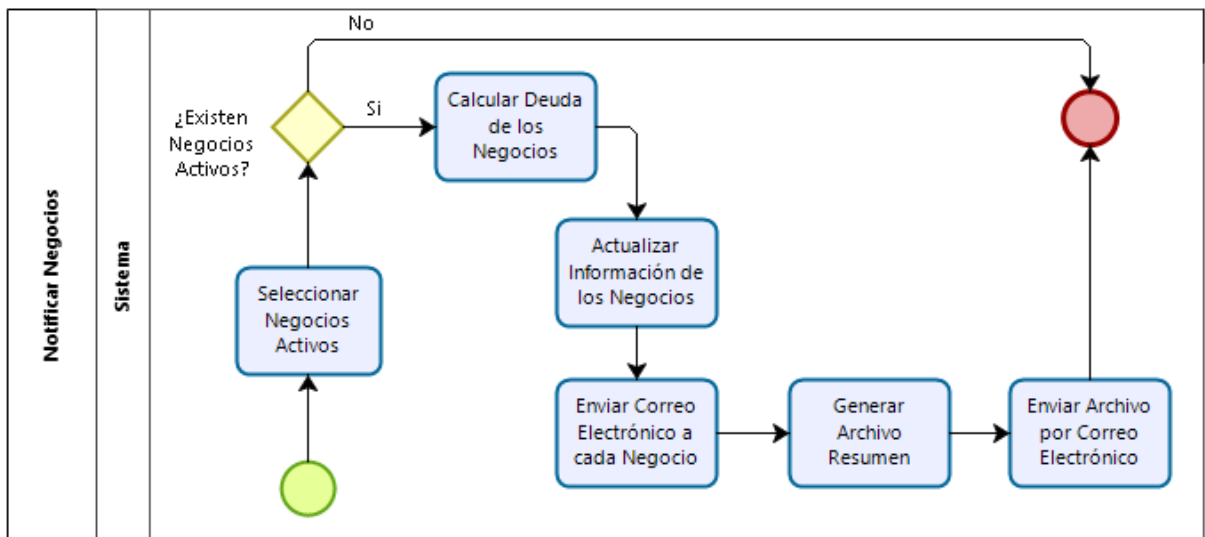
Diagrama de Actividades: Generar Estado de Cuentas



Fuente: Elaboración Propia.

Figura 24

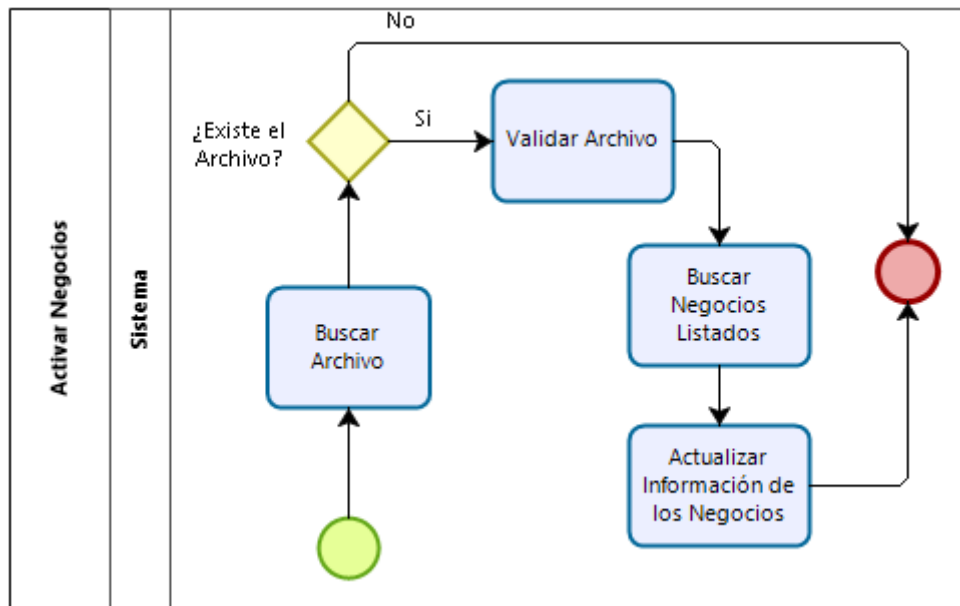
Diagrama de Actividades: Notificar Negocios



Fuente: Elaboración Propia.

Figura 25

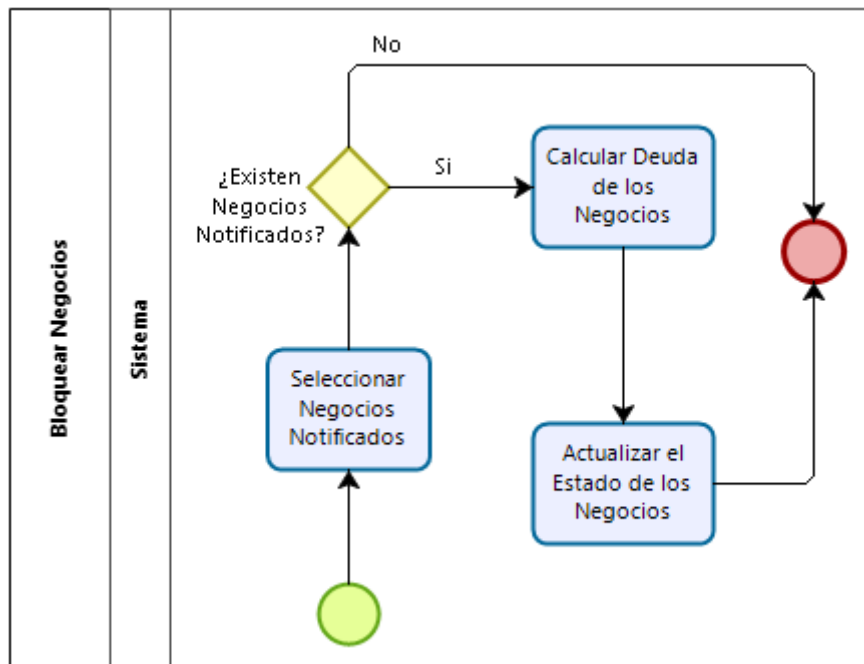
Diagrama de Actividades: Activar Negocios



Fuente: Elaboración Propia.

Figura 26

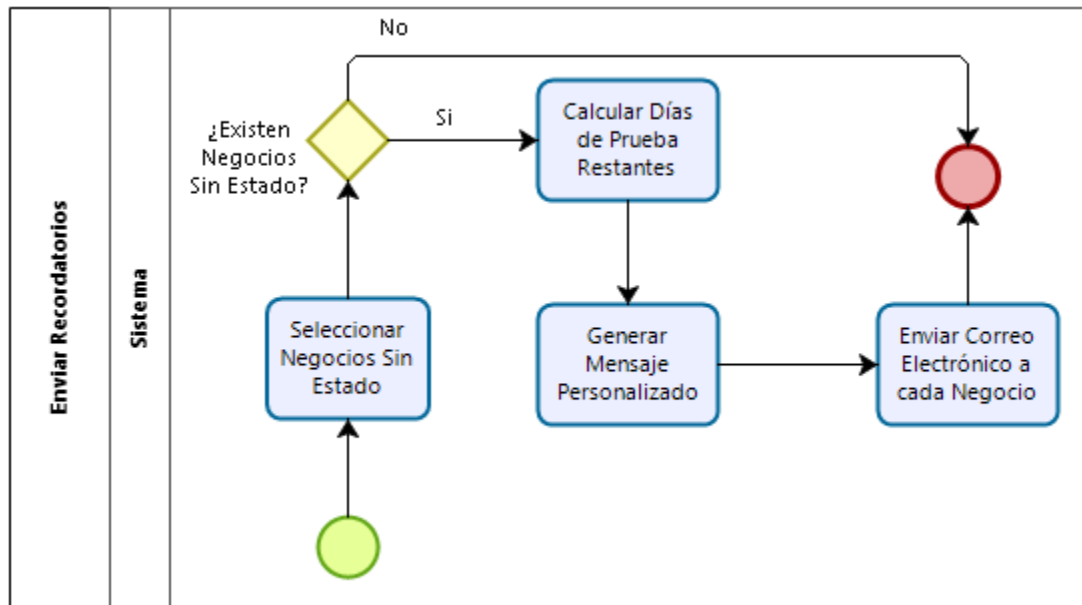
Diagrama de Actividades: Bloquear Negocios



Fuente: Elaboración Propia.

Figura 27

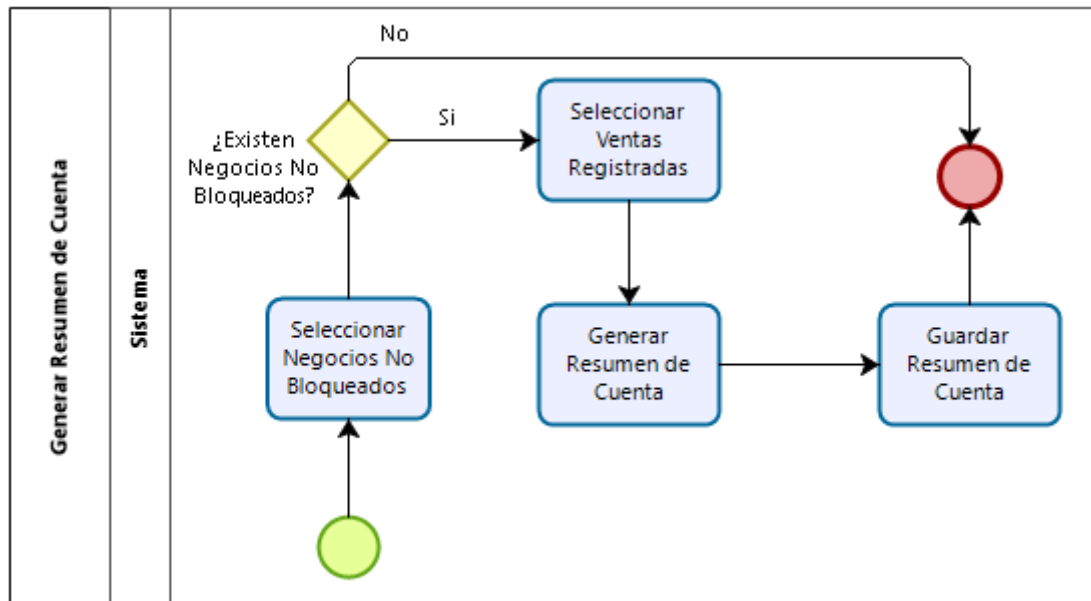
Diagrama de Actividades: Enviar Recordatorios



Fuente: Elaboración Propia.

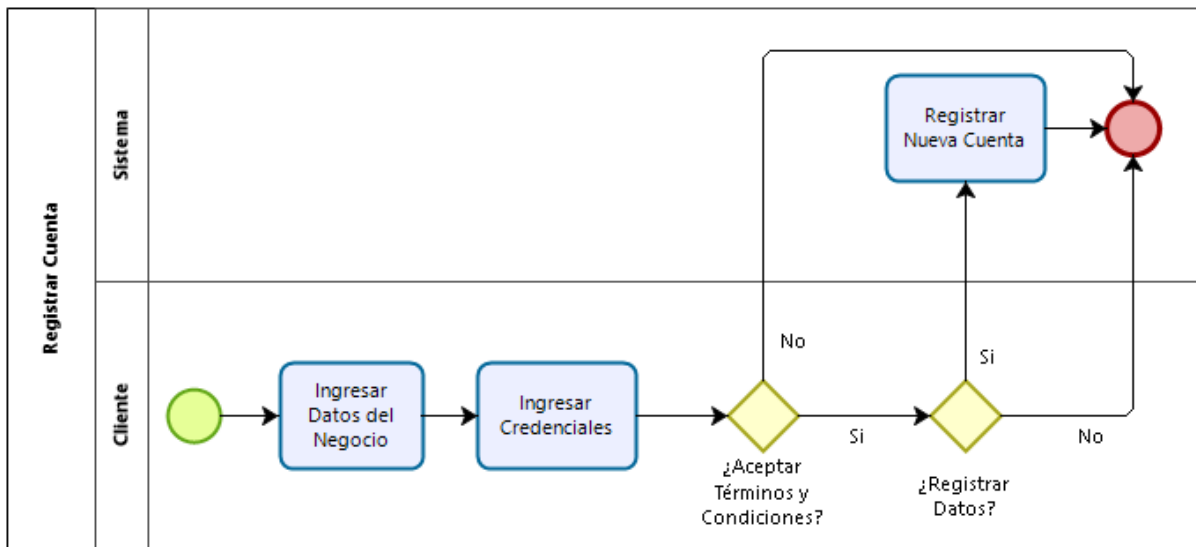
Figura 28

Diagrama de Actividades: Generar Resumen de Cuenta



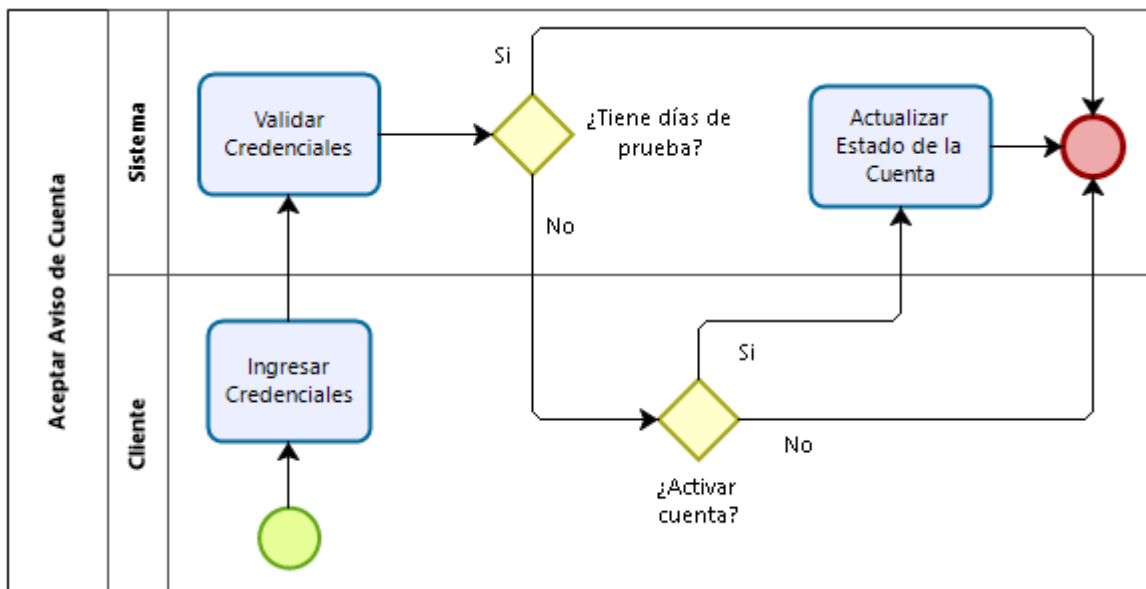
Fuente: Elaboración Propia.

Figura 29
Diagrama de Actividades: Registrar Cuenta



Fuente: Elaboración Propia.

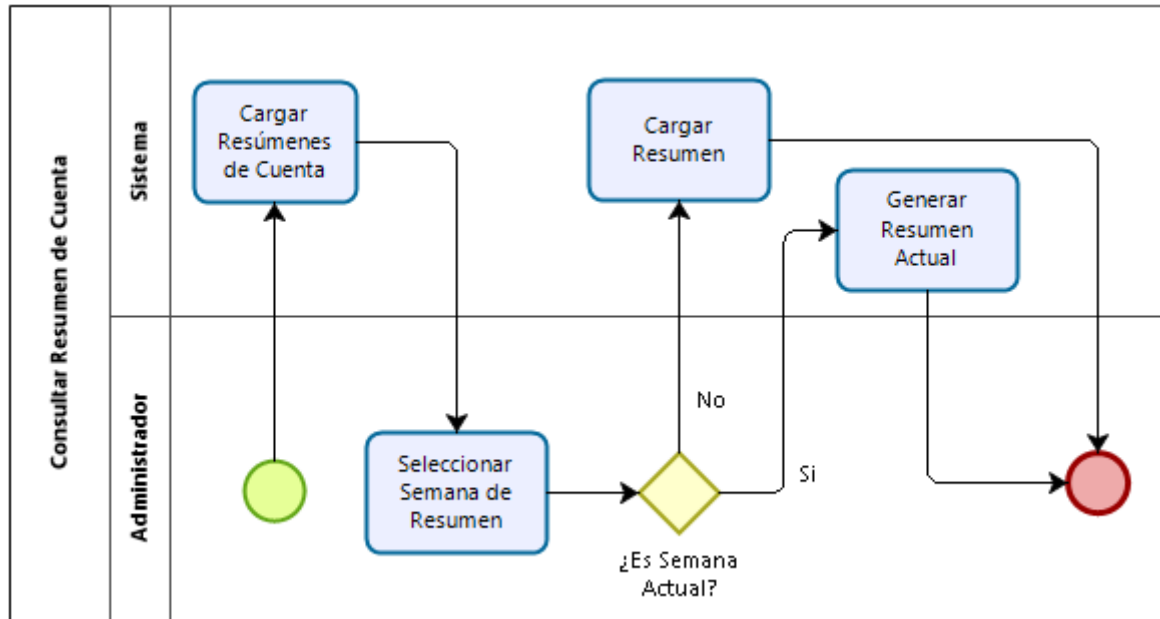
Figura 30
Diagrama de Actividades: Aceptar Aviso de Cuenta



Fuente: Elaboración Propia.

Figura 31

Diagrama de Actividades: Consulta Resumen de Cuenta



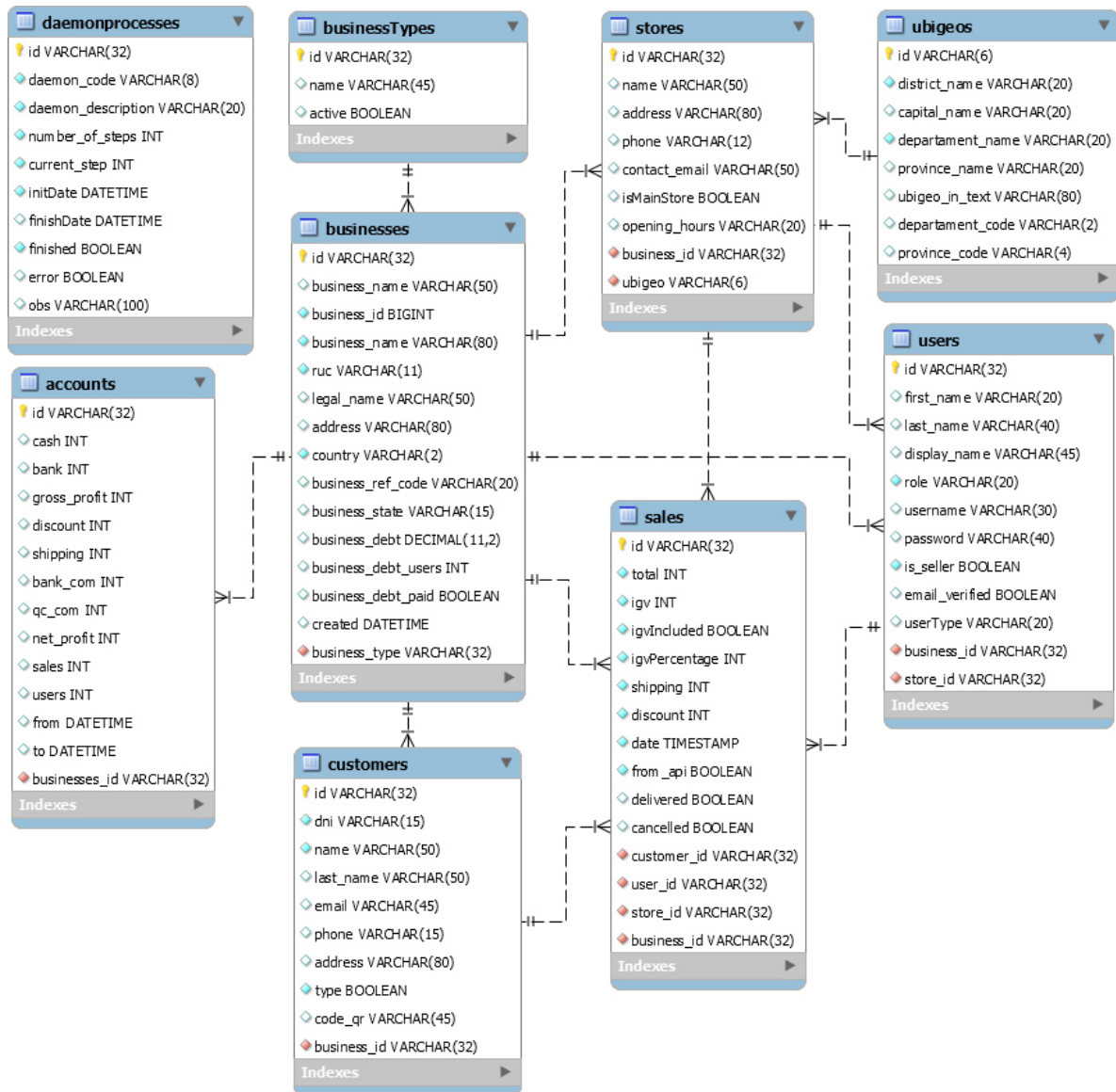
Fuente: Elaboración Propia.

En esta etapa también se llevó a cabo la modificación de la base de datos existente para que se pueda implementar el nuevo sistema (ver Figura 32).

Como colección central y principal se puede observar a “businesses” dicha colección tiene la información de los negocios que tiene el sistema, su estado, su estado de cuenta y otros datos (ver Figura 33).

La colección “users” tiene información de las cuentas usuarios que pertenecen a un negocio en específico, y según la cantidad de usuarios que tenga el negocio, se calculará el monto a pagar por concepto de suscripción al sistema.

Figura 32
Modelo de Datos



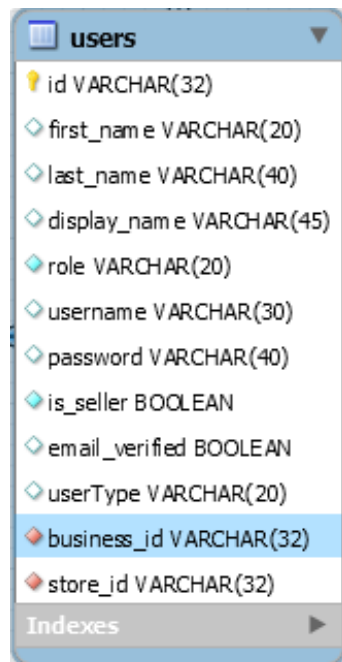
Fuente: Elaboración Propia.

Figura 33
Modelo de Datos - Colección businesses



Fuente: (Proyecto de Administración de Cuentas, 2018)

Figura 34
Modelo de Datos - Colección users



Fuente: (Proyecto de Administración de Cuentas, 2018)

IMPLEMENTACIÓN

En esta etapa se realizó el desarrollo del software, para llevarlo a cabo se utilizó el framework MEANJS, la arquitectura del servidor muestra una base de datos MongoDB junto con Node.js que tiene como dependencias a “mongoose” y “express”, por otra parte la arquitectura del cliente muestra Node.js y AngularJs. La comunicación entre el cliente y el servidor se realiza mediante un API RESTful implementado en el servidor.

Según la Metodología Scrum el desarrollo del proyecto debe realizarse en período cortos de tiempo o Sprints, por ello el desarrollo se dividió en 5 sprints.

Sprint 1

La primera tarea a realizar en el sprint 1 es desarrollar la arquitectura base del proyecto, para acelerar el proceso el equipo tomó como referencia una arquitectura ya desarrollada y la modificó para que soporte el nuevo proyecto.

La siguiente tarea fue implementar la autenticación de los usuarios o “login” (ver Figura 35), utilizando las credenciales tradicionales: correo electrónico y contraseña. La contraseña es encriptada y alojada en el servidor, mientras que por seguridad la sesión se cerrará automáticamente si hay una inactividad de 10 minutos por parte del usuario.

Figura 35

Vista Login



Fuente: (Proyecto de Administración de Cuentas, 2018)

Para los procesos automáticos o “demonios” (denominación según la cultura de la empresa) se utilizó la dependencia de Node.js llamada “cron”, que permite programar la ejecución de un proceso según la siguiente configuración:

* * * * *

Los seis asteriscos, de izquierda a derecha, significan:

- Segundo: Valores del 0 - 59.
- Minuto: Valores del 0 - 59.
- Hora: Valores del 0 - 23.
- Día del Mes: Valores del 1 - 31.
- Mes: Valores del 0 - 11 (enero - diciembre).
- Día de la Semana: Valores del 0 - 6 (domingo - sábado)

Entonces, para que el proceso automático se ejecute automáticamente el primer día de cada mes a las 00:00 horas, la configuración sería: 0 0 0 1 * *.

Finalmente, como resultado de este sprint se tiene el inicio de sesión de un usuario y la ejecución automática de un proceso que envía automáticamente notificaciones a las cuentas de Quipuc (ver Figura 36).

Figura 36
Notificación enviada al usuario Quipuc



Fuente: Elaboración Propia.

Sprint 2

En el sprint 2 se procedió al manejo de los estados de las cuentas Quipuc, los estados son 5 y, según las reglas de negocio, estas varían según la fecha y deudas que tenga la cuenta de Quipuc (a excepción de los estados “gratis” y “ninguno”), los estados son los siguientes:

- Ninguno (Cuenta Nueva).
- Activo.
- Notificado.
- Bloqueado.
- Gratis.

Figura 37
Negocios Notificados

Negocio	Dueño	Usuarios	Monto a Pagar	Pagado	Estado
<input type="text"/>	<input type="text"/>			Todo: ▾	Todos ▾
La Paella	Francisco	1	S/ 50.00	No	Notificado
Vanny	Vanessa	1	S/ 50.00	No	Notificado

Fuente: (Proyecto de Administración de Cuentas, 2018)

Como siguiente tarea era necesario el manejo de archivos de hojas de cálculo, para ello fue necesario utilizar la dependencia de Node.js llamada “exceljs”, que permite la generación y lectura de archivos de hojas de cálculo. Con esta dependencia se pudo generar un archivo de hoja de cálculo con los datos de todos los negocios de Quipuc, este archivo se adjunta al correo electrónico y se envía al usuario actual del sistema (ver Figura 38).

Figura 38
Correo Electrónico con Lista de Negocios adjunto



Fuente: Elaboración Propia.

Luego, se procedió a desarrollar la carga de archivos de hojas de cálculo que contiene como datos a los negocios que han cancelado sus deudas y es necesario cambiar de estado a “Activo” a dichos negocios.

Finalmente, era necesario el desarrollo de un proceso automático se ejecute automáticamente todos los días excepto el primer día del mes ya que el primer día de cada mes se ejecuta el proceso automático de notificaciones, por lo que su configuración de tiempo sería el siguiente: 0 0 0 2-31 * *. La finalidad de este proceso automático es leer el archivo previamente subido al servidor y cambiar de estado a “Activo” a los negocios que se encuentren en la lista (ver Figura 39).

Figura 39
Negocio Activado

Negocio	Dueño	Usuarios	Monto a Pagar	Pagado	Estado
<input type="text"/>	<input type="text"/>			Todo: ▾	Todos ▾
La Paella	Francisco	1	S/ 50.00	Si	Activo
Vanny	Vanessa	1	S/ 50.00	No	Notificado

Fuente: (Proyecto de Administración de Cuentas, 2018)

Sprint 3

La tarea principal del sprint 3 fue la de desarrollar el último proceso automático, este proceso debe ejecutarse todos los días 17 de cada mes, por lo que su configuración de tiempo sería el siguiente: 0 0 0 17 * *. La finalidad de este proceso automático es de cambiar el estado de los negocios de “Notificado” a “Bloqueado” (ver Figura 40). Por regla de negocio, el estado “Bloqueado” impide que el usuario del aplicativo Quipuc pueda acceder a ella hasta que vuelva a ser activada (manual o automáticamente)

Figura 40
Negocio Bloqueado

Negocio	Dueño	Usuarios	Monto a Pagar	Pagado	Estado
<input type="text"/>	<input type="text"/>			Todo: ▾	Todos ▾
La Paella	Francisco	1	S/ 50.00	Si	Activo
Vanny	Vanessa	1	S/ 50.00	No	Bloqueado

Fuente: (Proyecto de Administración de Cuentas, 2018)

El desarrollo para listar los negocios fue sencillo, sin embargo al existir una cantidad considerable de datos, fue necesario reducir el tamaño de los mismos mediante la reducción de nombres en los elementos de un Json (ver Figura 41).

Figura 41
Ejemplo de Reducción de Datos

```
Negocio {
  "id": "QID01",
  "nombreNegocio": "La Paella",
  "nombreDueño": "Francisco",
  "correo": "lapaella@gmail.com",
  "registro": "2016-10-07",
  "telefono": "963213554",
  "usuarios": 1,
  "monto": 50,
  "pago": true,
}

Negocio {
  "id": "QID01",
  "n": "La Paella",
  "d": "Francisco",
  "c": "lapaella@gmail.com",
  "r": "2016-10-07",
  "t": "963213554",
  "u": 1,
  "m": 50,
  "p": true,
}
```

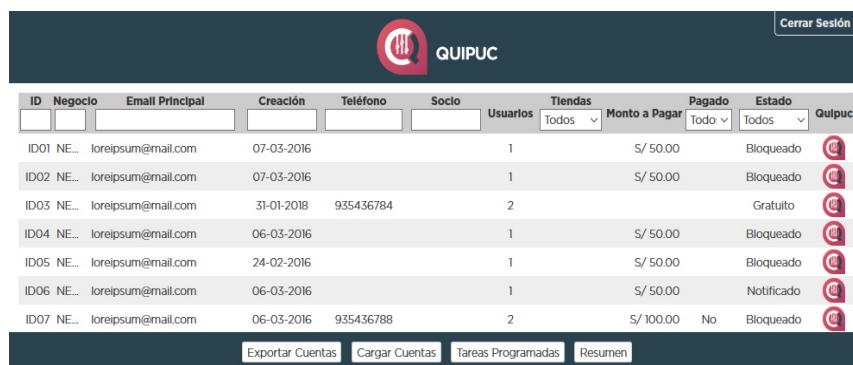
Fuente: Elaboración Propia.

Por ejemplo, el objeto de izquierda de la Figura 42 tiene un total de 200 caracteres, mientras que el objeto de la derecha tiene un total de 145 caracteres, una diferencia de 55 caracteres (27.5% menos), lo que significa menor carga de envío entre el servidor y el cliente. La reducción es más notoria mientras más elementos tenga el objeto Json o si se trata de una lista.

Con la aplicación de dicha estrategia se logró reducir la cantidad de data enviada significativamente, lo que se traduce en menor tiempo de carga, mejorando el tiempo de respuesta del servidor. Sin embargo, al reducir el nombre, es posible de perder la noción de qué representa cada elemento del Json, por lo que es bueno evaluar muy bien cuándo aplicar dicha estrategia.

Figura 42

Vista General del Sistema de Administración de Cuentas



ID	Negocio	Email Principal	Creación	Teléfono	Socio	Usuarios	Tiendas	Monto a Pagar	Pagado	Estado	Quipuc
ID01	NE...	loreipsum@mail.com	07-03-2016			1	Todos	S/ 50.00	Todo	Bloqueado	
ID02	NE...	loreipsum@mail.com	07-03-2016			1	Todos	S/ 50.00	Todo	Bloqueado	
ID03	NE...	loreipsum@mail.com	31-01-2018	935436784		2	Todos		Todo	Gratuito	
ID04	NE...	loreipsum@mail.com	06-03-2016			1	Todos	S/ 50.00	Todo	Bloqueado	
ID05	NE...	loreipsum@mail.com	24-02-2016			1	Todos	S/ 50.00	Todo	Bloqueado	
ID06	NE...	loreipsum@mail.com	06-03-2016			1	Todos	S/ 50.00	Todo	Notificado	
ID07	NE...	loreipsum@mail.com	06-03-2016	935436788		2	Todos	S/ 100.00	No	Bloqueado	

Fuente: (Proyecto de Administración de Cuentas, 2018)

Sprint 4

Este sprint se concentra en el desarrollo para acceder al aplicativo Quipuc desde el sistema, para esto se tuvo que coordinar la implementación con el proyecto Quipuc para que mediante un token autogenerado, el usuario del sistema pueda acceder a los datos de cualquier negocio en Quipuc.

Figura 43
Acceso a Quipuc



Fuente: Elaboración Propia.

Figura 44
Edición de estado

The image shows a form titled 'La Paella' with the ID 'QIDO1'. The email address is 'lapaella@gmail.com'. The owner is 'Francisco' and the phone number is '963213554'. The creation date is '07-10-2016'. There is one registered user and the amount to be paid is 'S/ 50.00'. At the bottom, there is a dropdown menu currently showing 'Activo'. Below the dropdown are two buttons: '✓ Guardar' (Save) and '× Cancelar' (Cancel).

Fuente: (Proyecto de Administración de Cuentas, 2018)

El token autogenerado hace que el usuario del sistema, pueda ingresar al Quipuc de un negocio seleccionado, pudiendo realizar soporte directamente como un usuario de Quipuc dentro del negocio.

La siguiente tarea es la de editar la información de estado de los negocios manualmente, para ello era necesario un acceso directo, realizar el cambio deseado y guardar (ver Figura 44).

Sprint 5

En este último sprint se desarrolló la tarea de agregar más campos a la edición de los datos del negocio en el sistema (ver Figura 45).

La siguiente tarea es realizar operaciones dentro del aplicativo Quipuc, respecto al nivel de permiso o rol que se va a utilizar, se optó por el rol de “Dueño”, para tener control absoluto de la cuenta Quipuc, pero según reglas del negocio, solo sería previa autorización y en casos en los que el usuario de Quipuc requiera soporte ante un problema o solicite una inducción al sistema.

Figura 45

Nuevos Campos en Edición



Fuente: (Proyecto de Administración de Cuentas, 2018)

PRUEBAS

Etapa donde las pruebas funcionales fueron realizadas al software desarrollado en la etapa de implementación, las pruebas se realizaron en base a los casos de prueba establecidos en la etapa de Análisis y Diseño. El documento resultante de esta etapa es el documento de pruebas, resultado de ejecutar cada caso de prueba y obtener un resultado (deseado o incorrecto).

Tabla 15

Trazabilidad de Casos de Prueba - Requisitos

	PBI-0020 PBI-0021 PBI-0022	PBI-0023 PBI-0024 PBI-0025	PBI-0030	PBI-0040 PBI-0041	PBI-0050 PBI-0051	PBI-0060	PBI-0070	PBI-0100	PBI-1000 PBI-1010 PBI-1020
CDP-1	X								
CDP-2		X							
CDP-3			X						
CDP-4				X					
CDP-5					X				
CDP-6						X			
CDP-7							X		
CDP-8								X	
CDP-9									X

Fuente: (Proyecto de Administración de Cuentas, 2018)

DESPLIEGUE Y MANTENIMIENTO

El despliegue del proyecto se realizó en los servidores de Amazon mediante su AWS EC2, creando un servidor para el nuevo proyecto y realizando la configuración necesaria para que se pueda desplegar sin problemas el proyecto. El servidor también soporta GIT por lo que el mantenimiento se realiza mediante ese medio; si existiese alguna actualización disponible para el proyecto, se detiene el sistema, se actualiza mediante GIT y se vuelve a

iniciar. Sin embargo, la actualización se debe realizar en horarios de entre las 11:00 pm y las 4:00 am.

La arquitectura consiste en una instancia de AWS EC2 que aloja al Sistema de Administración de Cuentas y otra instancia que aloja a la Base de Datos MongoDB (ver Figura 46). Ambos servidores se comunican internamente de forma local, lo que reduce considerablemente los tiempos de respuesta. La seguridad está a cargo de AWS, además, en caso de alta concurrencia, cuenta con un balanceador de carga como parte del servicio.

Figura 46

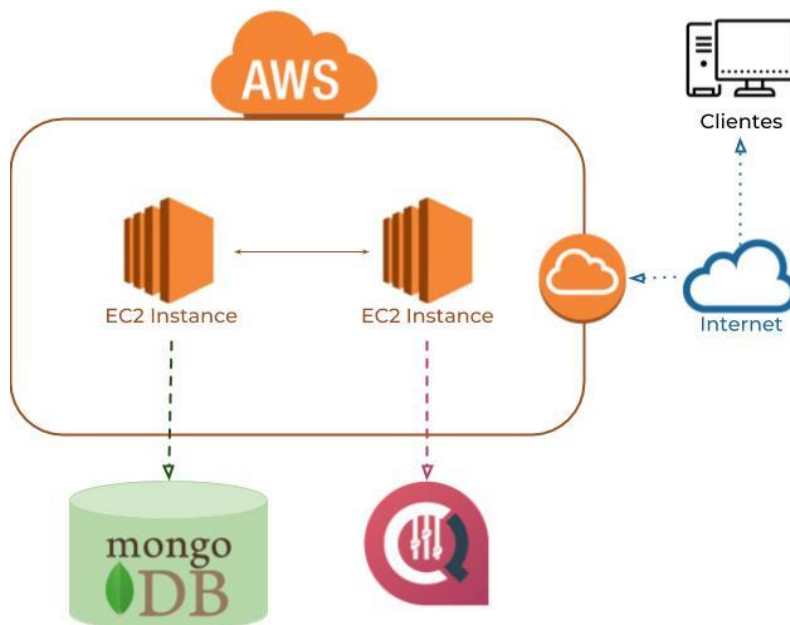
Instancias de Servidores AWS EC2

<div> <div>Launch Instance</div> <div>Connect</div> <div>Actions</div> </div>						
<div> <div>search : us-west-2b</div> <div>Name : quipuc</div> <div>Add filter</div> </div>						
<input type="checkbox"/>	Name	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
<input checked="" type="checkbox"/>	quipuc-admin	t2.micro	us-west-2b	running	2/2 checks ...	None
<input type="checkbox"/>	quipuc-mongo-master	t2.medium	us-west-2b	running	2/2 checks ...	OK

Fuente: (AWS EC2 Console, 2018)

Figura 47

Arquitectura del Servidor del Sistema de Administración de Cuentas



Fuente: Elaboración Propia.

3.3 EVALUACIÓN

3.3.1 EVALUACIÓN ECONÓMICA

COSTOS

Al ser un proyecto “In House”, los costos incurridos por el desarrollo del proyecto fueron asumidos directamente por la empresa. Entonces evaluamos los costos totales que la empresa ha realizado con recursos propios para ejecutar el proyecto (ver Tabla 16).

Estos costos se basan en los roles que han intervenido en el proyecto en alguna etapa del mismo. El cálculo se basa en el costo por hora de tener a dicho rol ejecutando una tarea en el proyecto, siendo el rol con mayor costo el de “Jefe de Proyecto” con un 65.31% del costo total del proyecto, esto se debe a que estuvo presente desde la concepción del proyecto hasta la finalización del mismo. El “Analista Funcional” cuenta con un costo del 23.95% y el “Analista Programador” un costo del 7.26%, responsables del desarrollo del proyecto. El “Tester” solo participa en la parte final del proyecto cuando se ejecutan los Casos de Prueba y por último de Diseñador que no tuvo mucha participación en el proyecto con un 0.87% de participación.

Tabla 16

Costos por Rol del Proyecto

Rol	Costo / Hora	Horas Trabajadas	Total Costo	Costo (%)
Jefe de Proyecto	S/ 22,50	320	S/ 7.200,00	65,31
Analista Funcional	S/ 12,00	220	S/ 2.640,00	23,95
Analista Programador	S/ 10,00	80	S/ 800,00	7,26
Tester	S/ 12,00	24	S/ 288,00	2,61
Diseñador	S/ 12,00	8	S/ 96,00	0,87

Total Horas	Total Proyecto
652	S/ 11.024,00

Fuente: Elaboración Propia.

Para alojar el servidor en la nube se utilizó Amazon Web Service EC2, creándose dos instancias: “quipuc-admin” con tipo de servicio “t2.micro” y “quipuc-mongo-master” con tipo de servicio “t2.medium” (ver Figura 46). Mantener activo dichos servicios tiene un costo por hora en dólares según el tipo de servicio creado (ver Figura 48).

Figura 48
Costos AWS EC2

Linux

RHEL

SLES

Windows

Windows con SQL Standard

Windows con SQL Web

Región: EE.UU. Oeste (Oregón) ▾

	CPU virtual	ECU	Memoria (GiB)	Almacenamiento de instancias (GB)	Uso de Linux/UNIX
Uso general – Generación actual					
t2.nano	1	Variable	0,5 GiB	Solo EBS	0,0058 USD por hora
t2.micro	1	Variable	1 GiB	Solo EBS	0,0116 USD por hora
t2.small	1	Variable	2 GiB	Solo EBS	0,023 USD por hora
t2.medium	2	Variable	4 GiB	Solo EBS	0,0464 USD por hora
t2.large	2	Variable	8 GiB	Solo EBS	0,0928 USD por hora
t2.xlarge	4	Variable	16 GiB	Solo EBS	0,1856 USD por hora

Fuente: (AWS EC2 Console, 2018)

Con la información obtenida, se realizó un cálculo total de lo cuesta mantener activo los servicios, para el tipo de cambio de Soles a Dólares Americanos se tomó como referencia a la información brindada por el Banco Central de Reserva del Perú, la información final se puede observar en la siguiente tabla:

Tabla 17
Costos por Servidor

Instancia	Tipo Instancia	Costo / Hora		Costo Anual	
quipuc-admin	t2.micro	USD	0,0116	USD	101,616
quipuc-mongo-master	t2.medium	USD	0,0464	USD	406,464

Total Dólares	USD	508,080
Tipo de Cambio	S/	3,332
Total Soles	S/	1692,92

Fuente: Elaboración Propia.

Respecto a gastos por energía eléctrica, servicios como internet, mobiliario, útiles de oficina, equipos informáticos, su instalación y configuración, la empresa no realizó gasto alguno, debido a que el personal trabajó de manera remota en sus propios hogares.

En resumen, los gastos incurridos durante la ejecución del proyecto, despliegue y mantenimiento se pueden observar en la siguiente tabla:

Tabla 18
Resumen de Gastos

Concepto	Costo Fijo	Costo Anual
Desarrollo del Proyecto	S/ 11.024,00	S/ 0,00
Servidores	S/ 0,00	S/ 1.692,92
Mobiliario	S/ 0,00	S/ 0,00
Otros (Internet, Luz Eléctrica, etc)	S/ 0,00	S/ 0,00

Fuente: Elaboración Propia.

INGRESOS

El proyecto tiene un ingreso estimado de S/ 550.00 mensuales sin impuestos en base a un total de 11 usuarios que van a tener, en conjunto, los 7 negocios que confirmaron continuar utilizando el aplicativo a pesar de que este deje de ser gratuito. Además se plantea el desarrollo de tiendas virtuales ya implementadas, lo supone un gasto simple de configuración e instalación en servidores propios o privados, con un ingreso estimado de S/ 400.00.

Tabla 19
Ingresos Estimados

Negocios	1376	Costo / Usuario	S/ 50,00
Confirmados	7	Cant. Usuarios	11
Por Confirmar	262	Total Usuarios	S/ 550,00
Pendientes	1082		
Gratuitos	25		
		Costo Tienda Virtual	S/ 100,00
		Cant. Tiendas Virtuales	4
1 Usuario	4	Total Tiendas Virtuales	S/ 400,00
2 Usuarios	2		
3 Usuarios	1	Total Ingreso (Sin IGV)	S/ 950,00

Fuente: Elaboración Propia.

3.3.2 INTERPRETACIÓN DEL VAN Y DEL TIR

VALOR ACTUAL NETO (VAN)

Según los gastos estimados del proyecto, la inversión para la realización del proyecto sería de: S/ 10664.00. Entonces se evalúan los ingresos que van a generar al tener implementado y operativo el proyecto (ver Tabla 20).

$$\text{Flujo de Caja Anual} = \text{Ingresos} - \text{IGV} - \text{Servicios}$$

$$\text{Flujo de Caja Anual} = 7788.00 - 1188.00 - 1692.92$$

$$\text{Flujo de Caja Anual} = 4907.08$$

Tabla 20

Flujo de caja proyectado

	Año 1	Año 2	Año 3	Año 4	Año 5
Flujo de Caja	4907,08	4907,08	4907,08	4907,08	4907,08

Fuente: Elaboración Propia.

Respecto a los montos anteriores, el beneficio del proyecto, en un plazo de 5 años, sería el de S/ 14,721.24 sin considerar impuestos. Pero este beneficio no sería real porque no se está considerando el valor del dinero en el tiempo, por lo que cada período esta tiene un valor menor ya que tenemos que tener en cuenta la tasa de descuento (la tasa de rentabilidad mínima que se espera ganar).

En base a la fórmula del Valor Actual Neto (VAN) en 5 años se recuperará la inversión realizada teniendo como tasa de descuento el 12%, que es el porcentaje de rentabilidad que suelen utilizar la empresa como mínimo para sus proyectos, logrando un saldo positivo de S/ 6,664.93:

$$VAN = -11024 + \frac{4907.08}{1 + 0.12} + \frac{4907.08}{(1 + 0.12)^2} + \frac{4907.08}{(1 + 0.12)^3} + \frac{4907.08}{(1 + 0.12)^4} + \frac{4907.08}{(1 + 0.12)^5}$$

$$VAN = 6,664.93$$

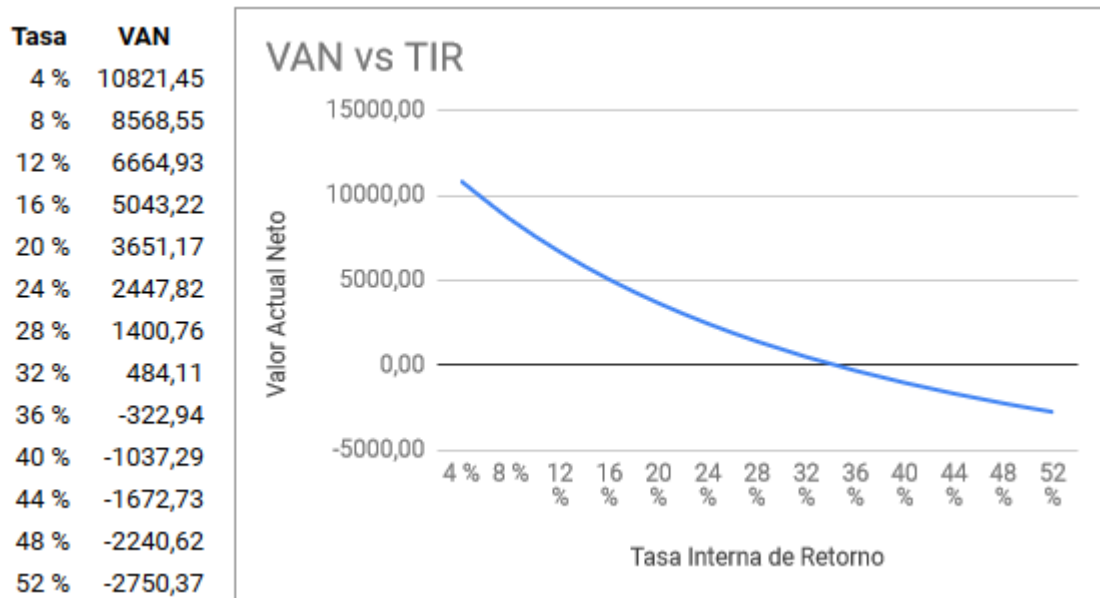
TASA INTERNA DE RETORNO (TIR)

Para evaluar la Tasa de Retorno Interno, se tiene que evaluar tendríamos que utilizar la fórmula del VAN, pero en este caso el valor del VAN sería 0, con esto se estaría hallando el valor de la tasa de descuento (ver Figura 49).

$$VAN = 0 = -11024 + \frac{4907.08}{1+i} + \frac{4907.08}{(1+i)^2} + \frac{4907.08}{(1+i)^3} + \frac{4907.08}{(1+i)^4} + \frac{4907.08}{(1+i)^5}$$

Figura 49

VAN vs TIR del Proyecto



Fuente: Elaboración Propia.

Mediante una hoja de cálculo, el VAN es igual a 0 cuanto la tasa alcanza el valor de: 34,3392 %. Si la tasa fuera mayor, el proyecto empezaría a dejar de ser rentable, ya que el monto invertido en el proyecto superaría ampliamente a los ingresos esperados. Sin embargo, a menor tasa, el proyecto se hace más rentable, ya que los ingresos superan lo invertido.

CAPÍTULO IV – REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

- La participación del autor del presente informe en el proyecto de Administración de Cuentas usando Scrum en la empresa RUQU S.A.C. fue el rol de Analista Funcional con especialidad en Sistemas Backend.
- El equipo estuvo conformado por cinco personas, todos pertenecientes a la empresa. En primer lugar tenemos al Jefe de Proyectos, encargado de la concepción, monitoreo y despliegue del proyecto, por la parte del equipo tenemos al Diseñador y Tester encargados de la parte inicial (prototipado y diseño) y final (ejecución de los casos de prueba) del proyecto respectivamente; también tenemos al Analista Funcional y Analista Programador, encargados del desarrollo del proyecto.
- El autor, con rol de Analista Funcional, tenía como responsabilidad principal la implementación del proyecto además de cumplir el cronograma establecido. Además, tenía otras responsabilidades como la configuración del ambiente de desarrollo, armado de la arquitectura, configuración de un ambiente de pruebas y despliegue del proyecto.
- El desarrollo bajo la metodología Scrum ayudó a que el equipo estuviera mejor comunicado y controlar mejor el desarrollo del proyecto ya que se tenía una mejor visibilidad cumpliendo así con los tiempos estimados en el cronograma. Esto ayudó también a presentar mejores entregables al “Product Owner” cada vez que había una reunión con el mismo, cumpliendo sus expectativas y corrigiendo algunos errores de interpretación de requerimientos en caso estos existiesen.

- Optar por una arquitectura de 3 capas donde la parte dirigida al cliente (Frontend) se comunica mediante un API RESTful con la parte del servidor (Backend) ayuda a la escalabilidad del proyecto, ya que al implementar una mejora o algún cambio en alguna parte (Frontend o Backend) no tendrá mucho impacto. Además, ante una eventual migración de tecnologías o cambios de entornos, el proyecto podrá mitigar la mayor cantidad de problemas de migración que surjan, disminuyendo así el tiempo de implementación de los cambios.
- Al participar en el proyecto, el autor tuvo la oportunidad de conocer más a detalle la administración de cuentas de un modelo de negocio de tiendas virtuales, las partes que la componen y la importancia de la seguridad para el acceso a cuentas.

CAPÍTULO V – CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se crearon los procesos automáticos necesarios para mantener los estados de las cuentas actualizadas según los requerimientos y lógica de negocio de la empresa.
- Se creó el proceso automático para notificar adecuadamente el costo por suscripción generado por cuenta.
- Se crearon accesos directos en el sistema para la ejecución manual de cualquier proceso automático en caso falle durante su ejecución o sea necesario ejecutarlo previa autorización.
- Se permitió controlar mejor las cuentas Quipuc, actualizando datos de las empresas que los procesos automáticos no pueden, monitoreando los procesos y los estados de los negocios.
- Se permitió controlar una cuenta Quipuc sin necesidad de conocer las credenciales. Sin embargo, este acceso está restringido y solo puede realizarse previa autorización del dueño de la cuenta.

5.2 RECOMENDACIONES

- El equipo de trabajo se puede adaptar muy bien con la metodología ágil Scrum, ya que tiene como fortaleza la comunicación y el trabajo en equipo, sin embargo es necesario controlar adecuadamente las reuniones debido a que pueden surgir rencillas entre miembros del equipo.
- El proyecto tenía un aspecto simple y de muy fácil elaboración, sin embargo a medida que se iba desarrollando y realizando cambios a pedido del Product Owner, este se empezó a tornar en algo más complejo.
- Reutilizar módulos o componentes ya implementados pueden ayudar a desarrollar más rápido las tareas asignadas, sin embargo, es recomendable de que éstas sean genéricas, ya que de lo contrario pueden resolver los problemas al corto plazo pero al momento de realizar pruebas con más data, pueden ocasionar comportamientos no deseados.
- Si bien es cierto que la metodología Scrum tiene muchas ventajas, es mejor adaptarla a la necesidad de la empresa y del proyecto al que se va a aplicar, además muchas empresas utilizan, dentro de sus procesos, artefactos y documentación de otras metodologías como por ejemplo los famosos Casos de Uso de la Metodología RUP.
- Tener en cuenta que la metodología Scrum al ser una metodología ágil tiende a no tener mucha documentación del proyecto, siendo esta una de sus desventajas, por lo que se recomienda al equipo separar un tiempo de desarrollo para la documentación. Ésta ayudará a que la curva de aprendizaje no sea tan elevada para las personas que se integren al equipo y/o tengan por encargo realizar mantenimiento del proyecto desarrollado.
- Cuando se detecte un problema en el sistema, desplegado un ambiente de Desarrollo o Producción, se recomienda establecer comunicación con el o los involucrados, esto permitirá poder entender mejor el problema, replicarlo de ser necesario y solucionarlo de la mejor forma.

- No tener algunos elementos disponibles al momento de desarrollar módulos que requieran dichos elementos (plantillas de informes, plantillas de hojas de cálculo, plantillas de correos electrónicos, etc.), pueden ocasionar un retraso significativo en el proyecto.

5.3 FUENTES DE INFORMACIÓN

PMIlatam. (28 de 08 del 2018). *Qué es la dirección de proyectos PMI*. Obtenido de <https://americalatina.pmi.org/latam>

SCRUMstudy. (12 de 09 del 2018). *Why SCRUM, Principles, Phases and Processes*. Obtenido de <https://www.scrumstudy.com>

SCRUMorg. (21 de 09 del 2018). *What is SCRUM?* Obtenido de <https://www.scrum.org/>

Idento. (10 de 09 del 2018). *¿Qué es una API REST?* Obtenido de <https://www.idento.es/blog/desarrollo-web>

RESTAPItutorial. (08 de 09 del 2018). *What is REST?* Obtenido de <https://www.restapitutorial.com/>

MEANJS. (16 de 09 del 2018). *Overview MEANJS.org*. Obtenido de <http://meanjs.org/docs.html>

AngularJS. (16 de 09 del 2018). *AngularJS*. Obtenido de <https://angularjs.org/>

ExpressJS. (16 de 09 del 2018). *ExpressJS*. Obtenido de <http://expressjs.com/>

MongoDB. (16 de 09 del 2018). *The MEAN Stack: MongoDB, ExpressJS, AngularJS and Node.js*. Obtenido de <https://www.mongodb.com/blog/>

Node.js. (16 de 09 del 2018). *About Node.js*. Obtenido de <https://nodejs.org/en/about/>

5.4 GLOSARIO

ORGANIZACIÓN ORIENTADA A PROYECTOS

Es un tipo de estructura de una organización en el cual existen pocos jefes de áreas pero existen muchos jefes de proyectos encargados de un grupo de trabajo, cada equipo, liderado por el jefe de proyectos, está encargado de la ejecución, implementación o desarrollo de un proyecto en específico. Es un tipo de estructura muy dinámica ya que al concluir un proyecto, los elementos del equipo quedan libres y pueden ser reasignados a otros proyectos.

APLICATIVO HÍBRIDO

Un tipo de aplicativo que se puede ejecutar en entornos web y entornos móviles, la gran ventaja de este tipo de aplicaciones es el único desarrollo que se realiza para desplegarlo en ambos entornos, reduciendo significativamente el tiempo de desarrollo que hubiese significado implementarlo en ambos entornos.

METODOLOGÍA ÁGIL

Son aquellas metodologías que permiten adaptar la forma de trabajo a lo que el proyecto necesita, consiguiendo flexibilidad y reducción de tiempos al implementar un proyecto. Gracias a las metodologías ágiles se consigue gestionar sus proyectos de forma eficaz, reduciendo costos e incrementando la productividad.

PRODUCT BACKLOG

Son los requerimientos del sistema, escrito en lenguaje de alto nivel que el Product Owner elabora y actualiza. El Product Backlog incluye requerimientos funcionales, no funcionales y, si así se considera, requerimientos técnicos por parte del equipo.

SPRINT

Parte de la metodología Scrum, un Sprint es una iteración de trabajo y es un tiempo fijo de desarrollo en el que se planea desarrollar un conjunto de tareas. El tiempo de duración se fija al inicio del proyecto, que como mínimo puede ser de una semana hasta un máximo de 30 días, según el equipo u organización lo requieran.

PRODUCT BACKLOG ITEM

Son las tareas priorizadas obtenidas del Product Backlog, un conjunto de estas se desarrollan durante una iteración de trabajo o Sprint. Si algún Product Backlog Item es demasiado grande (tiempo de desarrollo mayor a 16 horas), se fracciona en partes manejables para su desarrollo.

IDEMPOTENTE

Proceso o acción que, sin importar el número de veces que se ejecute o solicite, siempre termina o resulta obteniéndose un mismo valor que si se hubiese ejecutado una sola vez.

CUENTA QUIPUC

Una Cuenta Quipuc es un concepto que agrupa a 2 entidades, la entidad “Negocio”, representada por la colección “businesses” de la base de datos, y la entidad “Usuario” asignada al “Negocio”, representada por la colección “users”, la cantidad va desde una entidad a más (varios usuarios).

Ambas entidades, “Negocio” y “Usuario”, son lo básico para que la Cuenta Quipuc exista como tal.

DEMONIO

Según la cultura del negocio, un “demonio” es un proceso automático que se ejecuta periódicamente en un intervalo fijo del tiempo (cada hora, cada día, cada mes, etc.). La ejecución de estos procesos normalmente no requieren supervisión manual, sin embargo, es importante monitorearlos en caso ocurra un problema durante su ejecución o esté obteniendo resultados no deseados.

ANEXOS

ANEXO 1

Acta de Constitución

1. TÍTULO Y DESCRIPCIÓN DEL PROYECTO

“Proyecto de creación de Sistema de Administración de Cuentas para el Proyecto QUIPUC”

El objetivo principal de este proyecto es crear un sistema que logre controlar el acceso de las cuentas al aplicativo Quipuc. Dicho control deberá ser llevado a cabo de manera automática, brindando o restringiendo el acceso al sistema según ciertas condiciones establecidas como reglas de negocio.

2. DIRECTOR DE PROYECTOS ASIGNADOS Y NIVEL DE AUTORIDAD

Cesar Zegarra Medina será el jefe del proyecto y manejará los recursos asignados al mismo. Él tiene el poder de coordinar las tareas que delegará a los miembros del equipo, ejecutar el presupuesto, canalizar solicitudes de cambio, obtener la aprobación de los entregables.

3. CASOS DE NEGOCIO

El proyecto busca controlar el acceso de los usuarios al sistema Quipuc, logrando de esta manera generar ingresos adicionales a la empresa por el uso de la aplicación tanto web como móvil.

El proyecto también necesita tener acceso al sistema Quipuc para brindar algún soporte a un usuario o realizar demostraciones cuando se esté promocionando Quipuc.

4. RECURSOS PREASIGNADOS

Como primer recurso contamos con el director de proyectos que coordinará las actividades del proyecto. El jefe del proyecto tiene asignados 4 personas que conformarán el equipo del proyecto, y ocuparán roles asignados por el jefe del proyecto.

Los recursos que se necesiten deberán ser identificados en el proceso de planificación por el jefe del proyecto.

5. INTERESADOS

Se adjunta un documento llamado “**Registro de Interesados**” con un listado de personas y/o representantes de organizaciones interesadas en este proyecto.

6. LOS REQUISITOS DE LOS INTERESADOS HASTA LA FECHA

Se adjunta un documento llamado “**Historias de Usuario**” con un listado de requerimientos que se recopilaban en reuniones previas con el Product Owner.

7. DESCRIPCIÓN DE PRODUCTOS - ENTREGABLES

- Arquitectura
- Cronograma
- Alcance del proyecto

8. OBJETIVOS MEDIBLES DEL PROYECTO

El proyecto debe cumplir con lo planificado en el cronograma, en el cual se estima que el proyecto se va a completar en 1 mes y medio con los recursos propios de la empresa. El proyecto contará con un presupuesto estimado de S/ 12,000 Soles.

Estos objetivos deben ser corroborados en la planificación.

9. REQUISITOS DE APROBACIÓN DEL PROYECTO

- Gestionar el alcance del proyecto para que cumpla con los requisitos propuestos.
- Gestionar el tiempo estimado para la planificación del proyecto.
- Gestionar el costo estimado del proyecto.

10. RIESGOS DE ALTO NIVEL

Riesgo	Impacto	Probabilidad
El Product Owner pueda no disponer de tiempo en las reuniones pactadas con el equipo para mostrar el avance del proyecto. Sin embargo, también es posible modificar la hora de reunión para no perjudicar reuniones externas que pueda tener el Product Owner.	Baja	Baja

Mientras el proyecto se desarrolla, puede surgir un proyecto nuevo o uno ya existente que requiera la atención urgente de la empresa, lo que algún elemento del equipo deberá ser reasignado y dejar incompleto su trabajo.	Alto	Alta
Problemas de comunicación entre los miembros del equipo de proyecto, ya que algunos elementos disponibles de la empresa no han trabajado con el jefe proyectos anteriormente o entre sí.	Alto	Media
Algún elemento del equipo podría dejar de laborar en la empresa por lo que éste debe ser reemplazado por otra persona lo que el nuevo elemento tendría una curva de aprendizaje según cuanto esté avanzado el proyecto.	Alto	Baja
Durante la ejecución del proyecto, el equipo podría tener problemas personales o de salud que pudiera afectar el normal desarrollo del proyecto.	Alto	Media

11. PATROCINADORES DEL PROYECTO

El patrocinador de este proyecto es la misma empresa.

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 2

Registro de Interesados

Nombre del Interesado	Rol	Área	Interés en el Proyecto
Dennis Torres Wong	Director General Producto Owner	Dirección General	ALTA
Cesar Zegarra Medina	Jefe de Proyectos Web	Área de Proyectos Web	ALTA
Equipo de trabajo	Equipo de trabajo	Área de Proyectos Web	ALTA
DISNOVO	Externo	Externo	ALTA

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 3

Historias de Usuario

- Vamos a cobrar ex-post, el último del día de cada mes se le mandará la factura con su estado de cuenta tipo Google. (Cierre: 3pm)
- Se calcula el pago que debe hacer el Comercio por sus usuarios (prorratedos por día hasta la fecha de cierre). Si algún usuario ha sido desactivado por el Comercio, también debe calcular el monto de pago prorrateado por los días activos.
- El último día de cada mes se enviará una notificación del pago de los comercios. Y al admin master se le envía por email un excel con el estado de cuenta del mes de cierre (Nombre del Negocio, Email del Dueño, Estado de Cuenta, Estado de Pago - Completado con “No”).
- Cada vez que se suba ese excel actualizando los estados de pago se volverá a enviar una notificación a los comercios que no hayan pagado. Las políticas indican que debe hacerse mínimo en 2 fechas: día intermedio, un día previo al corte.
- Vamos a trabajar con una factura con pago a 15 días. Eso quiere decir que a más tardar el 15 de cada mes el cliente o su distribuidor deberán haber pagado su factura del periodo anterior. El 16 a medianoche, si la factura no ha sido cancelada, el sistema automáticamente desactiva al merchant: si se loguea le aparece un pop up que tiene una pendiente y en su tienda online le aparece la página 404.
- El demonio recorre el último excel cargado todas las madrugadas para reactivación de cuentas. Tener presente que el excel creado al fin de cada mes (Cierre) no interfiere con este archivo, las referencias son del mes anterior.
- El usuario activo gratuito, son usuarios a los cuáles no se les hace cobros porque son usuarios de prueba o comerciales para presentación y demo.
- Vamos a poner un campo a la hora de registrarse que se llame Código de Referido. Este código vinculará al merchant con el distribuidor. (Registro - Paso 2)

- Como buena práctica comercial el Distribuidor debería crear los usuarios de sus clientes con su propio código, además de hacerles la configuración que el cliente necesitará.
- En la pantalla de cobro del merchant se mencionara el nombre del distribuidor (socio).
- Se debe iniciar sesión para ingresar al panel de Admin Master. El usuario super admin será creado por base de datos el cual podrá ingresar de manera similar al dueño a todos los negocios en Quipuc.

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 4

Product Backlog

Código	Funcionalidad	Descripción
PBI-0010	Loguear a la aplicación.	
PBI-0020	Crear un demonio para que se ejecute mensualmente.	Crear un demonio para que se ejecute en un determinado día del mes (por defecto el día 1 a 00:00 horas de cada mes).
PBI-0021	Calcular la deuda, de cada negocio, según usuarios activos.	Calcular la deuda, de cada negocio, según usuarios activos y enviar un correo notificando la información de la deuda.
PBI-0022	Enviar una notificación con la información de la deuda.	
PBI-0023	Generar un excel con la lista de negocios junto con su respectiva deuda.	Generar un excel con la lista de negocios con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta, Estado de Pago; y enviarlo por email.
PBI-0024	Enviar por email el excel generado con la información de los negocios.	
PBI-0025	“Notificar” a los negocios.	Cambiar de estado a “notificado” a todos los negocios que tengan el estado 'activo'.
PBI-0030	Cargar un excel con información de los negocios.	Crear una tarea para la lectura de un excel con un formato definido (similar al formato mencionado anteriormente). El excel será guardado para un posterior uso.
PBI-0040	Crear un demonio para que se ejecute diariamente.	Crear un demonio para que se ejecute en una determinada hora de cada día (por defecto a las 00:00 horas).
PBI-0041	“Activar” a los negocios	Cambiar al estado “activo” a los negocios si cierta columna (del excel cargado anteriormente) tiene un valor definido para su activación.
PBI-0050	Crear un demonio para que se ejecute mensualmente.	Crear un demonio para que se ejecute en un determinado día del mes (por defecto el día 17 a 00:00 horas del mes en transcurso).
PBI-0051	“Bloquear” a los negocios.	Cambiar de estado a “bloqueado” a todos los negocios que tengan el estado 'notificado'.

PBI-0060	Exportar estados de cuenta.	Generar un excel con la lista de negocios con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta, Estado de Pago; y enviarlo por email.
PBI-0070	Listar Negocios.	Listar en la vista una tabla con las siguientes columnas: Nombre del Negocio, Email del Dueño, Estado de Cuenta y Estado de Pago, Nombre del Distribuidor, Código de Distribuidor (Editable).
PBI-0080	Buscar Negocio.	Permitir buscar un negocio en específico mediante su nombre o su email principal.
PBI-0090	Ingresar al Quipuc del negocio.	Ingresar automáticamente a la aplicación web de Quipuc correctamente autenticado. Los permisos deben ser de tipo “Dueño”
PBI-0100	Editar Negocio.	Mostrar el detalle del Negocio como: Nombre del Negocio, Email del Dueño, Estado de Pago, Nombre del Distribuidor, Estado de Cuenta y Código de Distribuidor; donde estos 2 últimos serán los únicos editables.
PBI-1000	Ingresar como “Super Admin”.	Para acceder a este modo, debe ser redirigido desde Quipuc Admin.
PBI-1010	Añadir un nuevo campo de referido al registrar un negocio.	Añadir un nuevo campo llamado “Código de Referido” que hará referencia al distribuidor. (Paso 2 del Registro).
PBI-1020	Realizar operaciones en Quipuc como dueño.	Manipular la información del negocio seleccionado en “Quipuc Admin” como rol “Dueño”.

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 5

Sprint Backlog

Sprint	Fecha Fin	Código	Descripción
Sprint 1	13/01/17	PBI-0010	Loguear a la aplicación.
		PBI-0020	Crear un demonio para que se ejecute mensualmente.
		PBI-0021	Calcular la deuda, de cada negocio, según usuarios activos.
		PBI-0022	Enviar una notificación con la información de la deuda.
Sprint 2	20/01/17	PBI-0023	Generar un excel con la lista de negocios junto con su respectiva deuda.
		PBI-0024	Enviar por email el excel generado con la información de los negocios.
		PBI-0025	“Notificar” a los negocios.
		PBI-0030	Cargar un excel con información de los negocios.
		PBI-0040	Crear un demonio para que se ejecute diariamente.
		PBI-0041	“Activar” a los negocios
Sprint 3	27/01/17	PBI-0050	Crear un demonio para que se ejecute mensualmente.
		PBI-0051	“Bloquear” a los negocios.
		PBI-0060	Exportar estados de cuenta.
		PBI-0070	Listar Negocios.
		PBI-0080	Buscar Negocio.
Sprint 4	02/02/17	PBI-0090	Ingresar al Quipuc del negocio.
		PBI-0100	Editar Negocio.
		PBI-1000	Ingresar como “Super Admin”.
Sprint 5	08/02/17	PBI-1010	Añadir un nuevo campo de referido al registrar un negocio.
		PBI-1020	Realizar operaciones en Quipuc como dueño.

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 6

Cronograma del Proyecto

	Nombre de la tarea	Duración	Inicio	Finalizar	Predecesoras
			<i>i</i> ▼		
1	[-] Proyecto Quipuc Admin	31d	09/01/17	20/02/17	
2	[-] Diseño	1d	09/01/17	09/01/17	
3	Diseño de la Arquitectura	1d	09/01/17	09/01/17	
4	Desarrollo de Prototipos	1d	09/01/17	09/01/17	
5	[-] Implementación	22d	10/01/17	08/02/17	2
6	Login	1d	10/01/17	10/01/17	
7	[-] Cobranza	11d	11/01/17	25/01/17	6
8	[-] Cobranza Mensual	5d	11/01/17	17/01/17	
9	Calcular deuda y Notificar	3d	11/01/17	13/01/17	
10	Generar Resumen	2d	16/01/17	17/01/17	9
11	Cargar Excel de Actualización	1d	18/01/17	18/01/17	8
12	[-] Actualización diaria	2d	19/01/17	20/01/17	11
13	Leer Excel de Actualización	1d	19/01/17	19/01/17	
14	Activar Negocios	1d	20/01/17	20/01/17	13
15	Bloquear Negocios	1d	23/01/17	23/01/17	12
16	Exportar Estados de cuenta	2d	24/01/17	25/01/17	15
17	[-] Gestión	4d	26/01/17	31/01/17	7
18	Listar Negocios	1d	26/01/17	26/01/17	
19	Buscar Negocio	1d	27/01/17	27/01/17	18
20	Editar Negocio	1d	30/01/17	30/01/17	19
21	Acceder al Negocio	1d	31/01/17	31/01/17	20
22	[-] Externo	6d	01/02/17	08/02/17	17
23	Acceder al Aplicativo Externo	2d	01/02/17	02/02/17	
24	Modificar Registro de Usuarios	2d	03/02/17	06/02/17	23
25	Acceder con Rol "Dueño"	2d	07/02/17	08/02/17	24
26	[-] Pruebas	6d	09/02/17	16/02/17	5
27	Pruebas Funcionales	2d	09/02/17	10/02/17	
28	Correcciones y modificaciones	4d	13/02/17	16/02/17	27
29	[-] Despliegue	2d	17/02/17	20/02/17	26
30	Configuración del servidor	1d	17/02/17	17/02/17	
31	Despliegue del aplicativo	1d	20/02/17	20/02/17	30

Fuente: Elaboración Propia.

ANEXO 7

Plan de Despliegue

1. Introducción

1.1. Propósito

El presente plan de despliegue se utilizará cuando el proyecto Quipuc Admin haya sido completamente desarrollado. Sin embargo, se puede utilizar para desplegar un ambiente de pruebas en caso el Director General así lo requiera.

1.2. Alcance

El presente documento solo es exclusivo para el proyecto Quipuc Admin, en lo que se refiere a despliegue. Será utilizado para preparar el ambiente donde se alojará el producto final de Quipuc Admin.

2. Plan de Despliegue

Se muestran los pasos a seguir para implementar el servidor de producción, son un conjunto de comandos a ejecutar en el terminal del servidor:

- `sudo apt-get update`
- `sudo apt-get install git`
- `wget -q0-
https://raw.githubusercontent.com/xtuple/nvm/master/install.sh | sudo bash`
- `sudo nvm i 6.9.2`
- `sudo nvm alias default 6.9.2`
- `sudo npm i -g pm2`
- `echo export PATH="$HOME/npm/bin:$PATH" >> ~/.bashrc`
- `npm config set prefix ~/npm echo`
- `"export
NODE_PATH=$NODE_PATH:/home/$USER/npm/lib/node_modules" >>
~/.bashrc && source ~/.bashrc`
- `nano ~/.bashrc`

```
alias sudo='sudo env PATH=$PATH:$HOME/npm/bin'
```

```
reopen putty client
```

- sudo apt-get install nginx
- sudo nano /etc/nginx/sites-available/default

Ejemplo:

```
server {  
  
    listen: 80;  
  
    server_name: admin.quipuc.com;  
  
    location: {  
  
        proxy_pass: http://172.xxx.xxx.xxx:8080;  
  
        proxy_http_version: 1.1;  
  
        proxy_set_header: Upgrade $http_upgrade;  
  
        proxy_set_header: Connection 'upgrade';  
  
        proxy_set_header: Host $host;  
  
        proxy_cache_bypass: $http_upgrade;  
  
    }  
  
}
```

- sudo service nginx restart
- sudo git clone
https://github.com/gitQuipuc/quipuc_admin.git
- cd quipuc_admin
- npm i
- sudo pm2 start server.js --name quipuc_admin --interpreter
/usr/local/nvm/v6.9.2/bin/node
- sudo service pm2 start quipuc_admin

Antes de ejecutar el último comando, es necesario crear el archivo “private.env”, copiar los valores del archivo “public.env” y modificarlos con variables de entorno según el servidor y referencias de producción.

3. Responsabilidades

El responsable directo de la ejecución de este Plan de Despliegue es el Jefe de Proyectos.

La conformidad de la ejecución del presente plan está a cargo del Director de Proyectos, quien verificará el correcto funcionamiento de la aplicación desplegada en el servidor.

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 8

Formularios de Reunión

Formulario Nro 1 - Fecha: 13/01/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none">- Se consiguió ahorrar tiempo al reutilizar la arquitectura de un proyecto anteriormente desarrollado.- El Inicio de Sesión ya está funcional, por lo que se pudo avanzar con otras tareas.- Al tener conocimientos del manejo de monedas, se pudo ahorrar tiempo en las pruebas.	<ul style="list-style-type: none">- Los servidores de prueba no estaban estables e impedían un desarrollo fluido al momento de realizar la presentación al Product Owner.- El módulo de envío de correos electrónicos no estaba funcionando correctamente debido a que no se elaboraron correctamente las plantillas que se utilizan.	<ul style="list-style-type: none">- Verificar el servidor de pruebas horas antes de la reunión con el Product Owner para realizar una buena presentación.- Verificar que información debe ser enviada al cliente en el correo electrónico para realizar validaciones previas a su envío.

Formulario Nro 2 - Fecha: 20/01/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none">- Se reutilizó un módulo ya elaborado para la lectura de archivos de cálculo.	<ul style="list-style-type: none">- Se tuvo que modificar la arquitectura seleccionada ya que al haber tareas automatizadas no existía un módulo para ellos.- La lectura y envío de archivos de cálculo tuvo una demora en su desarrollo debido a que las plantillas a utilizar no estaban elaboradas o no se habían enviado al equipo.	<ul style="list-style-type: none">- Revisar toda la arquitectura para verificar que no se presenten problemas de módulos.- Solicitar con mayor tiempo de anticipación las plantillas de los archivos a utilizar en el desarrollo.

Formulario Nro 3 - Fecha: 27/01/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none"> - La búsqueda de negocios se realizó en la mitad del tiempo estimado. - La exportación de estados de cuenta se logró en menor tiempo de lo estimado debido a que se tenía la plantilla a utilizar. - En coordinación con el proyecto Quipuc se logró avanzar con el acceso al negocio. 	<ul style="list-style-type: none"> - El listado de negocios está tomando mucho tiempo en completarse, posiblemente no se está realizando la consulta adecuadamente. - Se están bloqueando Negocios que no tienen un estado definido o "Ninguno". 	<ul style="list-style-type: none"> - Verificar si se están realizando consultas a la Base de Datos correctamente y si es posible optimizarlas. - Verificar las variables que se utilizan al momento de realizar consultas y actualizaciones cuando se realizan cambios de estados masivos en la base de datos.

Formulario Nro 4 - Fecha: 02/02/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none"> - La edición de negocios tomó menos tiempo de lo estimado. 	<ul style="list-style-type: none"> - El acceso al Quipuc del negocio tuvo demoras en la implementación y no se pudo terminar en el tiempo estimado. - No se esperaba problemas de compatibilidad en el momento de solicitar acceso al proyecto Quipuc. 	<ul style="list-style-type: none"> - Se optó por cambiar la estrategia de acceso. Se procederá a redirigir a la aplicación Quipuc y se identificará automáticamente con un token dinámico previamente generado.

Formulario Nro 5 - Fecha: 08/02/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none"> - Las operaciones del administrador como “Dueño” (rol dueño en Quipuc), se desarrollaron en menor tiempo de lo estimado. 	<ul style="list-style-type: none"> - El sistema tuvo varias observaciones al momento de ejecutar las pruebas. - Se agregó un cambio nuevo: Las tareas automáticas deben poder ser ejecutadas manualmente. 	<ul style="list-style-type: none"> - Corregir las observaciones.

Formulario Nro 6 - Fecha: 15/02/2017

¿Qué salió bien en la iteración?	¿Qué no salió bien en la iteración?	¿Qué mejoras vamos a implementar en la próxima iteración?
<ul style="list-style-type: none"> - Las observaciones fueron corregidas. - El sistema pasó las pruebas realizadas. - El despliegue se realizó de manera satisfactoria ya que el servidor ya se encontraba configurado anteriormente. 		

Fuente: Elaboración Propia.

ANEXO 9

Documento de Pruebas

1. INTRODUCCIÓN

1.1. Objetivo

El objetivo de este documento es recoger los casos de pruebas que verifican que el sistema satisface los requisitos especificados. Deberá contener la definición de los casos de prueba, la matriz de trazabilidad entre casos de pruebas y requisitos, y la estrategia a seguir en la ejecución de las pruebas.

1.2. Alcance

El presente documento solo es exclusivo para el proyecto Quipuc Admin, en lo que se refiere a pruebas. Será utilizado cuando en proceso de desarrollo culmine, para verificar que se cumple con los requerimientos establecidos.

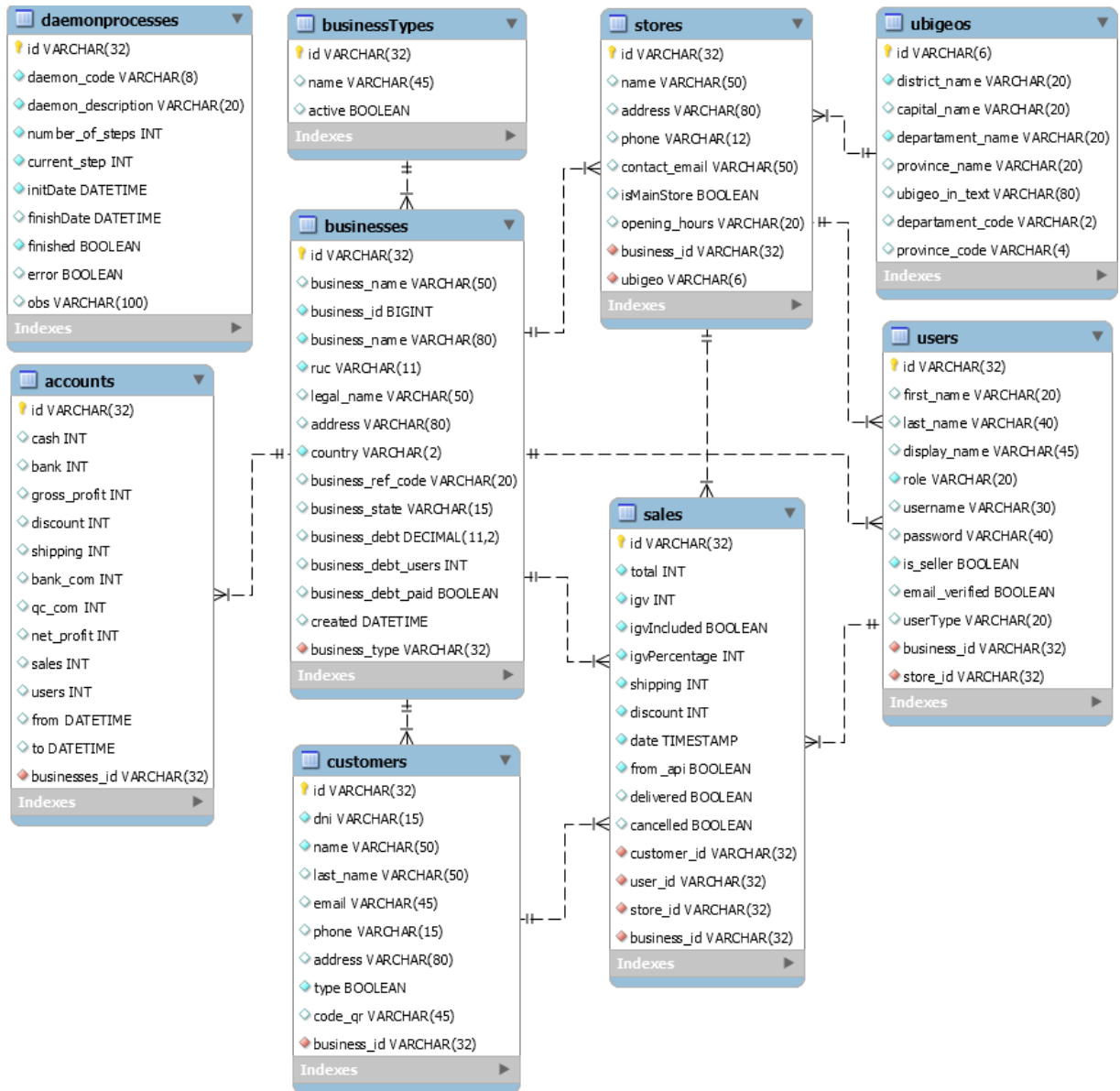
2. TRAZABILIDAD DE CASOS DE PRUEBAS - REQUISITOS

	PBI-0020 PBI-0021 PBI-0022	PBI-0023 PBI-0024 PBI-0025	PBI-0030	PBI-0040 PBI-0041	PBI-0050 PBI-0051	PBI-0060	PBI-0070	PBI-0100	PBI-1000 PBI-1010 PBI-1020
CDP-1	X								
CDP-2		X							
CDP-3			X						
CDP-4				X					
CDP-5					X				
CDP-6						X			
CDP-7							X		
CDP-8								X	
CDP-9									X

Fuente: (Proyecto de Administración de Cuentas, 2018)

ANEXO 10

Modelado de Datos



Fuente: Elaboración Propia.